

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ
«Компьютерлік технологиялар» кафедрасы

«Қорғауға жіберілді»
Кафедра меңгерушісі
ф.-м.ғ.д., проф. З.Қ. Құралбаев

_____ « ____ » _____ 2014 ж.
(КОЛЫ)

ДИПЛОМДЫҚ ЖОБА

Такырыбы: «Ағылшын тілін үйренуге арналған бағдарлама және Oxford 3000 сөздігі»

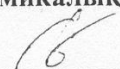
5B070400 – Есептеу техникасы және бағдарламалық қамтамасыз ету мамандығы бойынша

Орындаған: ВТк-10-2 Абдуллаев А.Б.

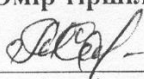
Жетекші: ф.-м.ғ.д., проф. Құралбаев З.Қ.

Кенесшілер :

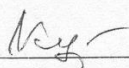
Экономикалық бөлім бойынша :

 « 28 » _____ доцент Боканова Г.Ш.
(КОЛЫ) 05 2014 ж.

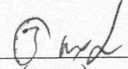
Өмір тіршілігі қауіпсіздігі бойынша:

 « 26 » _____ т.ғ.к., аға оқытушы Муташева Г.С.
(КОЛЫ) 05 2014 ж.

Есептеу техникасын қолдану бойынша :

 « ____ » _____ ф.-м.ғ.д., профессор Құралбаев З.Қ.
(КОЛЫ) 2014 ж.

Мөлшер бақылаушы:

 « ____ » _____ аға оқытушы Рахимжанова З.М.
(КОЛЫ) 2014 ж.

Пікір жазушы :

_____ « ____ » _____ АТУ, т.ғ.д., проф. Заурбеков Н.С.
(КОЛЫ) 2014 ж.

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Ақпараттық технологиялар» факультеті
«Есептеу техникасы және бағдарламалық қамтамасыз ету» мамандығы
«Компьютерлік технологиялар» кафедрасы

жобаны орындауға берілген

ТАПСЫРМА

Студент Абдуллаев Акмал Бердиярұлы

Жоба тақырыбы « Ағылшын тілін үйренуге арналған бағдарлама және Oxford 3000 сөздігі»

Ректордың « ___ » _____ № ___ бұйрығы бойынша бекітілген.

Аяқталған жұмысты тапсыру мерзімі: « ___ » _____ 2014 ж.

Жобаға қажетті алғашқы мәліметтер (талап етілетін жоба нәтижелерінің параметрлері) және нысананың бастапқы деректері:

MS SQL Server 2008 дерекқорларды басқару жүйесін және Embarcadero RAD Studio Delphi 2010 бағдарламалау ортасын пайдалану арқылы ағылшын тілін үйренуге ыңғайлы қосымша құрылды.

Диплом жобасындағы әзірленуі тиіс сұрақтар тізімі немесе диплом жобасының қысқаша мазмұны:

- Техникалық тапсыма.
- Ақпараттық жүйелер, дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.
- Бағдарламаның дерекқорын жобалау, жобалау құралдарын таңдау, ER диаграммасын жобалау, дерекқорды құру, бағдарламаның интерфейсін жобалау және оны дерекқормен байланыстыру.
- *Техникалық-экономикалық негізделуі.*
- Өндірістегі өміртіршілік қауіпсіздігі шараларын ұйымдастыру.
- Бағдарламаның жұмыс істеу принципі мен функционалдығын көрсететін графикалық материалдар.

Негізгі ұсынылатын әдебиеттер

- 1 Кандзюба С.П., Громов В.Н. Delphi 6/7. Базы данных и приложения. СПб: ООО «ДиаСофт», 2002.
- 2 Д. Петкович Microsoft SQL Server 2012. Руководство для начинающих. –БХВ-Петербург, 2013. – 816с.
- 3 Никита Культин. Основы программирования в Delphi 2010. – Санкт-Петербург, 2010. – 448с.
- 4 Мамаев Е. MS SQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: BHV, 2004. – 416с.
- 5 Марк Шпеник, Оррин Следж. Microsoft SQL Server 2000 DBA Survival Guide. – М.: Вильямс, 2001. – 236 с.

Жоба бойынша бөлімшелерге катысты белгіленген кеңесшілер

Бөлім	Кеңесші	Мерзімі	Қолы
Негізгі бөлім	Құралбаев З.Қ.		
Тіршілік қауіпсіздігі	Муташева Г.С.		
Экономикалық бөлім	Боканова Г.Ш.		
Норма бақылаушы	Рахимжанова З.М.		
Есептеу техникасын қолдану	Құралбаев З.Қ.		

ДИПЛОМ ЖОБАСЫН ДАЙЫНДАУ

КЕСТЕСІ

№ р/с	Тарау аттары, әзірленетін сұрақтардың тізімі	Жетекшіге ұсыну мерзімдері	Ескерту
1	Техникалық тапсырма		
2	Ақпараттық жүйелер, дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.		
3	Бағдарламаның дерекқорын жобалау, жобалау құралдарын таңдау, ER диаграммасын жобалау, дерекқорды құру.		
4	Дерекқормен пайдаланушы интерфейсін байланыстыру, қосымша жасау.		

Тапсырманың берілген уақыты « _____ » _____ 2014ж.

Кафедра меңгерушісі _____
(қолы) ф.-м.ғ.д., проф. Құралбаев З.Қ.

Жоба жетекшісі _____
(қолы) ф.-м.ғ.д., проф. Құралбаев З.Қ.

Орындалатын тапсырманы қабылдаған студент _____
(қолы) Абдуллаев А.Б.

Андатпа

Бұл дипломдық жобада ағылшын тілін үйренуге арналған бағдарлама және Oxford университеті ғалымдары зерттеген ағылшын тілінде ең көп қолданылатын сөздерден тұратын, ағылшынша-орысша-қазақша Oxford 3000 сөздігін құрастыру қарастырылған.

Бағдарламаның деректер қоры ДҚБЖ MS SQL Server 2008, интерфейсі Embarcadero RadStudio Delphi 2010 және дизайны Adobe Photoshop бағдарламалық құралында жасалған.

Бұл бағдарлама сөздік, арнайы сабақтар, кестелер, ағылшынша-қазақша мақал-мәтелдер, ағылшын тілінен білімін тексеруге арналған тесттерден тұрады.

Аннотация

В данной дипломной работе предусматривается разработка программа для изучения английского языка и англо-русско-казахский словарь состоящий из более распространенных 3000 слов в повседневной жизни выбранные учеными Оксфордского университета.

База данных программы разрабатывался в СУБД MS SQL Server 2008, а интерфейс и дизайн на программах Embarcadero RadStudio Delphi 2010 и Adobe Photoshop.

Это программа состоит из словаря, специальных уроков, таблиц, пословицы на англо-казахском языках, тестов для проверки уровня знания английского языка.

Annotation

In this diploma work includes the development of a program to learn English and the English- Russian - Kazakh dictionary consisting of the most common 3,000 words in everyday life chosen by scientists at Oxford University .

Database of program developed DBMS MS SQL Server 2008 , and the interface and design programs at Embarcadero RadStudio Delphi 2010 and Adobe Photoshop.

This program consists of a dictionary , special classes , tables, proverbs in the English- Kazakh , tests to check the level of knowledge of English language.

Мазмұны

Кіріспе	7
1 Техникалық тапсырма.....	8
1.1 Бағдарламаның тағайындалуы және қолдану аймағы.....	8
1.1.1 Техникалық құрылғыларға қойылатын талаптар	8
1.1.2 Бағдарламалық құжатқа қойылатын талаптар	8
1.2 Құру кезеңдері мен сатылары	8
1.2.1 Құру кезеңі бойынша орындалатын жұмыстардың мазмұны	9
1.2.2 Тестілеу	9
2 Ақпараттық жүйелер. Деректер қорын жобалау жүйесі(ДҚБЖ).....	10
2.1 Ақпараттық жүйелер туралы мағлұматтар	10
2.1.1 Ақпараттық жүйенің тіршілік циклі.....	13
2.1.2 Ақпаратты жүйені еңгізу сатысы	15
2.2 Деректер қорын басқару жүйесі.....	16
2.2.1 Деректер қорын басқару жүйесі	16
2.2.2 Дерекқордың жіктелуі	17
2.2.3 Деректер моделі.....	18
2.2.4 Дерекқорды басқару жүйесінің құрылымы мен функциялары	20
2.2.5 Microsoft SQL server ДҚБЖ	21
2.2.6 Кесте құру және өшіру.....	22
3 Деректер қорын жобалау	23
3.1 ДҚ құрудың алғашқы кезеңі.....	23
3.2 Концептуалды жобалау.....	23
3.3 ER моделін құру	23
3.3.1 Бизнес ережелер	24
3.3.2 Реляционды қатынастарды құрастыру.....	26
3.3.3 Логикалық жобалау.....	30
3.3.4 Көрсетілімдер және сұраныстар	32
3.4 Физикалық жобалау	33
3.4.1 Дерек қорына қосымша құру	33
3.4.2 Бағдарламаны сипаттау	34
3.5 Орнату дискісін құру	39
4 Тіршілік қауіпсіздігі.....	44
4.1 Жұмыс сипаттамасы	44
4.2 Жұмыс орнының микроклимат жағдайы.....	44
4.3 Бөлменің өрт қауіпсіздігі.....	44
4.4 Бөлменің өрт қауіпсіздігі және электр қауіпсіздігі	44
4.5 Шудың әсерін зерттеу.....	45
4.6 Бөлме жоспары	45
4.7 Кондиционерлеу жүйесін есептеу	46
4.8 Бөлменің вентиляция жүйесін есептеу	50
4.9 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды.....	52
5 Техника-экономикалық негізделуі	54

5.1 Жұмыстың сипаттамасы.....	54
5.2 Маркетингтік стратегия.....	54
5.2.1 Бағдарламаның қамтаманың өңделуінің еңбек өнімділігін есептелуі	55
5.2.2 Қосымша шығындар статьясы.....	57
5.2.3 Өзіндік күн нәтижесінің кестесі.....	61
5.2.4 Бағдарламаның бағасын есептеу.....	62
5.2.5 Кәсіпорынның бірмезгілдік шығындары.....	64
5.2.6 Бағдарламаны еңгізудегі алынған үнемдеу мөлшері мен табыс.....	65
5.2.7 Құрылған программаны орнату.....	64
5.3 Экономикалық тиімділікті есептеу.....	66
5.3.1 Таза ағымдағы құндылықты есептеу.....	66
5.3.2 Пайда индексін есептеу.....	67
5.3.3 Табыстың ішкі нормасын есептеу.....	67
5.3.4 Өтімділік периодын есептеу.....	68
5.4 Экономика бөлімі бойынша қорытынды.....	68
Қорытынды.....	69
Пайдаланылған әдебиеттер тізімі.....	70
А қосымшасы.....	71
Ә қосымшасы.....	73

Бұл дипломдық жобада ағылшын тілін үйренуге арналған бағарлама және қосымша Oxford 3000 сөздігі бағдарламасы жобаланды.

Бұл бағдарлама ағылшын тілін үйрену үшін өте ыңғайлы. Бағдарламада ағылшын тілі грамматикасы, ағылшын тілінен білім деңгейін тексеруге арналған тесттер, ағылшын тілінде ең көп қолданылатын етістіктер, ағылшынша-қазақша мақал-мәтелдер, күнделікті өмірде қолданатын, әр түрлі тақырыптағы ағылшынша-қазақша тілашар, және Оксфорд университеті ғалымдары зерттеуі бойынша алынған ағылшын тілінде ең көп қолданылатын 3000 сөзден тұратын ағылшынша-орысша-қазақша Oxford 3000 сөздігін жасау қарастырылған.

Дерекқор құру үшін алдымен талдау жасалынды және ең бірінші сөздіктің кестесімен ең көп қолданылатын етістіктердің кестесі құрылды одан кейін мақал-мәтелдер және тілашар кестесі құрылды. Тілашар үшін әр бір тақырыпқа жеке кесте жасалынды. Деректер қоры MS SQL Server-2008 жүйесінде жасалады. Бағдарламаның дерекқоры MS SQL Server дерекқорларды басқару жүйесінде құрылғаннан кейін, пайдаланушылардың барлық сұраныстарын қанағаттандыратын бағдарламаның интерфейсі жасалды. Бағдарламаның интерфейсі Embarcadero RAD Studio Delphi 2010 бағдарламалық құралында іске асырылды.

Бұл бағдарлама ағылшын тілін үйренушілер үшін ыңғайлы сонымен қатар, аз уақыт ішінде ағылшын тілін меңгеруге мүмкіндік береді.

1 Техникалық тапсырма

1.1 Бағдарламаның тағайындалуы және қолдану аймағы

Ағылшын тілін үйренуге арналған және Oxford 3000 сөздігі бағдарламасы. Бұл бағдарлама ағылшын тілін үйренуге арналған қазақ тіліндегі бағдарлама. Бағдарлама мынадай бөлімдерді қамтиды:

- ағылшын тілі грамматикасы;
- ағылшын тілінен білім деңгейін тексеруге арналған тест;
- Oxford 3000 сөздігі;
- ағылшын тілінде ең көп қолданылатын етістіктер;
- ағылшынша-қазақша мақал-мәтелдер;
- тілашар;
- ағылшын тілін үйренушілерге кеңес.

1.1.2 Техникалық құрылғыларға қойылатын талаптар

Бағдарламаны орнату үшін компьютердің сипаттамасы төмендегідей:

- процессор Pentium–1.7Hz;
- жедел жады 1Гигабайттан кем емес;
- HDD, 10 Гигабайт бос орын болуы керек;
- операциялық жүйе Windows 2000 Server, Windows 2003, XP, 7 және Windows 8;
- Microsoft SQL Server 2008 бағдарламалық құралы.

1.1.3 Бағдарламалық құжатқа қойылатын талаптар

Бағдарламалық құжаттың құрамыт төмендегідей болуы қажет:

- техникалық тапсырма;
- бағдарлама және байқаулардың әдістері;
- бағдарламаны қолданушыға көмек.

1.2 Құру кезеңдері мен сатылары

Құру сатылары

Құру келесі үш сатыны қамту керек:

- техникалық тапсырма жасау;
- бағдарламаны жобалау жұмыстары;
- ақпараттарды енгізу.

Құру кезеңдері

Құру жұмыстары техникалық тапсырмаға сай болуы керек. Құру кезеңдері төмендегідей болады:

- ақпараттар жүйесін құру;
- құжатты құру;
- бағдарламаның мысалмен тексерілуі.

1.2.1 Құру кезеңі бойынша орындалатын жұмыстардың мазмұны

Құру кезеңі бойынша келесі жұмыстар орындалу қажет:

- тапсырма жалпы мақсатын анықтау;
- бағдарламаны қолданатын техникалық құрылғыларға қойылатын талаптарды анықтау;
- жобаға қойылатын жалпы талаптарды анықтау.

1.2.2 Тестілеу

Тестілеу – бұл бағдарламадағы қателерді іздеу.

Қате – бұл бар нәрселердің сәйкес келмеуі.

Сенімділік – әрбір қабылдалмаудың құнын есепке алып есептелінетін, белгілі уақыт ішіндегі қарсылық жасалмаған жұмыс.

Тестілеу принциптері

Тестілеу жүйеде қалған табу үшін, сонымен қатар жүйенің сенімділігін және құндылығын арттыру үшін жасалады. Тестілеу арқылы дұрыс жобаланбаған жүйенің жоғарғы дәрежедегі сенімділігіне жете алмайды.

Егер бағдарламаға тестілеу жүргізетін болсақ, белгілі бір жағдайлармен бағдарламаның құнының көтеру арқылы тестілеуге кеткен барлық шығындарды қайтарып алуымыз керек. Мұны бағдарлама сенімділігін жоғарылатумен ғана жүргізуге болады. Сенімділікті тек енгізілген қателерді дұрыстау арқылы ғана жоғарлатуымызға болады.

Сәтті тест дегеніміз, қате тапқан тестке айтамыз. Егерде бір қате табылмайтын болса, онда ол сәтсіз тест болады.

- қателер топтасу қасиетіне ие;
- тестілеу әдісін жеңілдету үшін бағдарламаны өзгертпеңіз;
- егер бағдарлама жасаушы бағдарламаны жазуда қате жіберсе, онда оны тестілеуден өткізуде сол қатені жіберуі мүмкін;
- тестіні құру – бұл шығармашылық үрдіс, яғни кей жағдайларда мидың дұрыс жұмыс істеуін қажет етеді;
- қатені дұрыс табатын тест өте жақсы тест болып есептеледі;
- бағдарлама не қызмет атқарады, яғни тек өзіне тиесілі жұмысты істейді ме, әлде басқа жұмыстарды да істеуге мүмкіндігі барма соны тексеру қажет;
- комментариилердің аз болуы қателерді табуды қиындатады;
- комментариилердің аз болуы қатені табуды қиындатумен қатар, түсініктеме сөздігі болмаса қиын бағдарламаны талқылау да қиын болады.

Тестілеу біткен соң қате жоқ екендігіне кепілдік бере алмаймыз, тек белгілі бір деңгейде жүйенің жұмысы дұрыс екендігі жайлы айтуға болады.

Тест – кіріс деректерінің жиыны және жауап бағдарламаның іс–әрекеті болып есептеледі. Қарапайым бағдарламалармен жан–жақты тестілеу жасау мүмкін емес, өйткені оған уақыт және қажетті ресурстарда жеткіліксіз. Тестілеудің қатені іздеу мүмкіндігіне байланысты және тестті құру әдісіне байланысты төменде бірнеше түрі көрсетілген.

Тестілеу түрлері:

- құрылымдық тест жүргізу;
- функционалдық тестілеу.

Құрылымдық тестілеу

Бұл тестілеу барысында бағдарлама тексті ашық түрде тұрады. Шешімдердің циклдері және блоктары тестіленеді.

Құрылымдық тестілеу бірнеше түрге бөлінеді:

- операторларды жабу әдісі;
- шешімдерді жабу әдісі;
- шарттарды жабу әдісі;
- шарттарды аралас жабу;
- циклдерді тестлеу.

Жалпы түсініктер үшін құрылымдық тестілеудің мысалдарына тоқтап кетуге болады.

Функционалдық тестілеу

Функционалдық тестілеуде керісінше бағдарламалардың тексті көрінбейді, бағдарлама қара жәшік, яғни кіріс және шығыс шарттары белгілі болады, оған қоса жұмыстың толық структурасы қарастырылады. Бағдарламаның спецификациясы бойынша тестілеу жүргізіледі.

Бұл тестілеудің бірнеше түрі берілген:

- шекті мәндерінің анализі;
- жүйе эксплуатациясы;
- эквивалентті кластар;
- белгілі көлемдегі тестілеу;
- тәжірибелі эксплуатация;
- белгілі мөлшердегі тестілеу;
- қорғанысты тестілеу.

2 Ақпараттық жүйелер. Деректер қорын жобалау жүйесі (ДҚБЖ)

2.1 Ақпараттық жүйелер туралы мағлұматтар

Ақпараттық жүйе дегеніміз – қандай да бір пайдаланушы қызметіне арналған ақпараттық модель. Жүйе мынадай үрдістерді қамту керек:

- ақпаратты сақтау;
- өзара ақпарат ауысу;
- ақпаратты түрлендіру.

Қазір ақпараттық жүйелердің әр салада маңыздылығы артып келуде.

Ақпараттық жүйе көмегі арқылы орындалатын тапсырмалар әр түрлі болып келуінің арқасында құрылымы жағынан бір–бірінен айырмашылығы бар ақпараттық жүйелер әр түрлі болып бөлінеді.

Ақпараттық жүйелер әр түрлі белгілері бойынша бөлінеді. Негізгі белгісі, құрылу ерекшеліктерімен функционалдық мүмкіндігі бөлу болып есептеледі. Ақпараттық жүйелер қолданатын техникалық құралдарға байланысты және орындайтын тапсырмасына қарай әр түрлі сыныптарға бөлінеді (Сурет 2.1).

Ақпараттық жүйелер берілген ақпаратты сақтау жағынан фактографиялық және құжаттық деп екіге бөлінеді. Фактографиялық ақпараттық жүйелер мәтін және сандар түріндегі ақпаратты сақтап, өңдеуге арналған. Ал құжаттық ақпараттық жүйелерде ақпарат құжат түрінде болады.

Автоматтандырылған жүйелерге негізделіп, ақпараттық жүйелер автоматтандырылған, қолмен басқарылатын, және автоматтандырылмаған деп бөлінеді.



Сурет 2.1– Ақпараттық жүйелердің бөлінуі

2.1.1 Ақпараттық жүйелерді жобалау

Ақпараттық жүйелерді жобалау әрдайым жобаның мақсатын анықтаумен басталады. Кез келген сәтті жобаның негізгі міндеті жүйені іске қосқан кезе және оны пайдалану уақытында мыналарды қамтамасыз ету болып табылады:

- жүйенің талап етілген функциялығы және оның функциялануының құбылмалы жағдайларына бейімділік дәрежесін;
- жүйенің талап етілген өткізу қабілетін;
- сұранысқа жүйенің талап етілген реакция уақытын;
- талап етілген режимде жүйенің тоқтаусыз жұмысы, басқаша сөзбен айтқанда – қолданушылардың сұраныстарын өңдеу үшін жүйенің дайындығы мен жетімділігі;

- пайдалану қарапайымдылығы мен жүйенің қуаттауын;
- қажетті қауіпсіздікті.

Өнімділік жүйенің тиімділігін анықтайтын негізгі фактор болып табылады. Жақсы жобалық шешім жоғары өндіруші жүйенің негізі болады.

Ақпараттық жүйелерді жобалау үш негізгі саланы қамтиды:

- дерекқорда жүзеге асырылатын деректер объектіні жобалау;
- деректерге сұраныстардың орындалуын қамтамасыз ететін бағдарлама, экрандық нысан, есеп беруді жобалау;
- нақты орта немесе технологияны есепке алу, нақтырақ: желі топологиясы, аппараттық құралдардың конфигурациясы, пайдаланылатын архитектура (файл-сервер немесе клиент-сервер), параллельді өңдеу, деректерді үлестірілген өңдеу және т.б.

Жобалаудың шынайы жаңдайында – бұл белгіленген шектеулерді ескере отырып, бар технологиялардың құралдары жүйесінің функциялық талаптарын қанағаттандыратын тәсілді іздеу. Кез келген жобаға бірқатар абсолютті талаптар қойылады, мысалы жобаны әзірлеудің максималды уақыты, жобаға максималды қаржы құю және т.б. Жобалаудың қиындықтарының бірі ол жобаға қойылатын талаптарды талдау немесе сол не басқа жобалық шешімдерді жүзеге асыру секілді құрылымдалған міндет болып табылмайтындығында. Күрделі жүйені принцип бойынша сипаттау мүмкін емес деп есептеледі. Бұл мекемені басқару жүйелеріне қатысты. Негізгі дәйектерден бірі жүйенің функциялану жағдайын өзгерту болып табылады, мысалы жаңа басшылықтың сол немесе басқа ақпарат ағындарын директивті өзгертуі. Тағы бір дәйек – ірі жоба үшін жүздеген парақтардан тұратын техникалық міндеттердің көлемі, онда техникалық жобада қателікер болуы мүмкін.

Егер ұғынатын болсақ, жүйенің дамуын осылай болжап білуге болмай ма және ол туралы ақпарат алу расымен де мүмкін емес пе? Мүмкін, жүйе туралы және оның дамуының болжамдалатын (басшылықпен) жолдары туралы бүтіндей түсінікті семинарлардың көмегін алуға болар. Осыдан кейін жүйені қарапайым құрамдастарға бөлі, құрамдастар арасындағы байланысты ықшамдау, құрамдастардың еркіндігін қарастыру және олардың арасындағы интерфейстерді сипаттау (бірі құрамдастың өзгеруі автоматты түре басқа құрамдастың өзгеруіне әкелмес үшін), сонымен қатар жүйені кеңейту мүмкіндіктері мен функция жүйенің сол немесе басқа нұсқасында іске асырылмау үшін «бітеуіштер». Осындай элементарлы ұғымдардан келіп шығып, ақпараттық жүйеде іске асырылуы болжанып отырған сипаттама шынайыға ұқсайды. Ақпараттық жүйелерді жасаудың классикалық тәсілдерін ұстануға болады, олардан бірі «сарқырама» схемасы – төменде сипатталған. Ақпараттық жүйелерді әзірлеуде кейбір басқа тәсілдер қысқаша қарастырылады. Төменде сипатталған қай тәсілді ұстану кереу (және өзіндік тәсіл ойлау табудың мәні бар ма) – қандай да бір өлшемде талғам мен мән-жайдың ісі. Бағдарламалық жасақтаманың өміршеңдік кезеңі оны құру және пайдалану үлгісінен тұрады. Үлгі осы БЖ да қажеттілік туғаннан бастап, оның

барлық қолданушыларда қолданыстан толық шығу уақытымен аяқталатын оның әртүрлі жағдайларын бейнелейді.

2.1.2 Ақпараттық жүйенің тіршілік циклі

Ақпараттық жүйенің тіршілік циклі жүйенің әр түрлі күйін көрсетеді. Оның алғашқы қажеттілігенен бастап толық пайдаланып біткенге дейін қарастырады.

Каскадтық үлгі

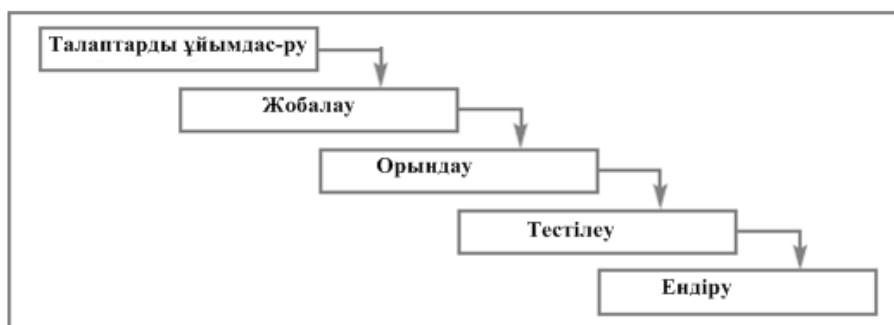
Келесі кезеңге ауысу алдыңғы кезеңде жұмыстарды толық аяқталуын білдіреді (Сурет 2.2).

Аралық бақылаумен кезең бойынша үлгісі

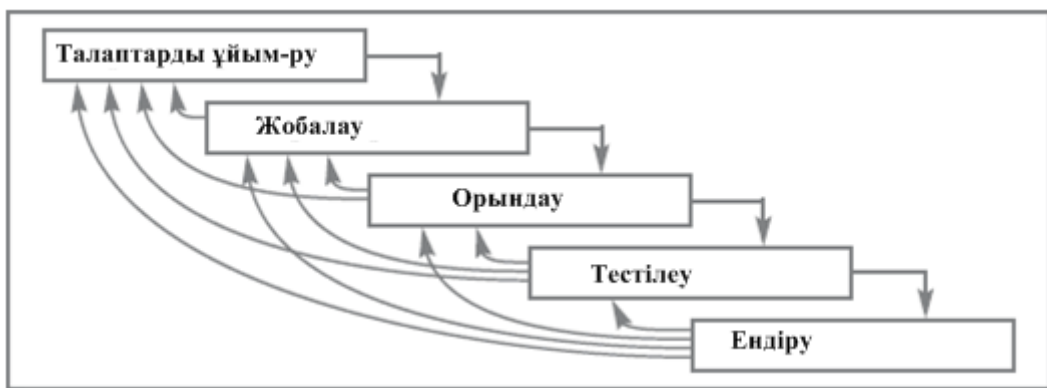
БЖ ны әзірлеу кезеңдер арасында кері байланыс циклдарымен итерациялармен жүреді. Кезеңаралық түзетулер каскадтық үлгімен салыстырғанда әзірлеу процесінің еңбек жұмсалымын азайтуға мүмкіндік береді; әр кезеңнің өміршеңдік уақыты әзірлеудің бүкіл кезеңіне созылады (Сурет 2.3).

Спиральды үлгі үлгі

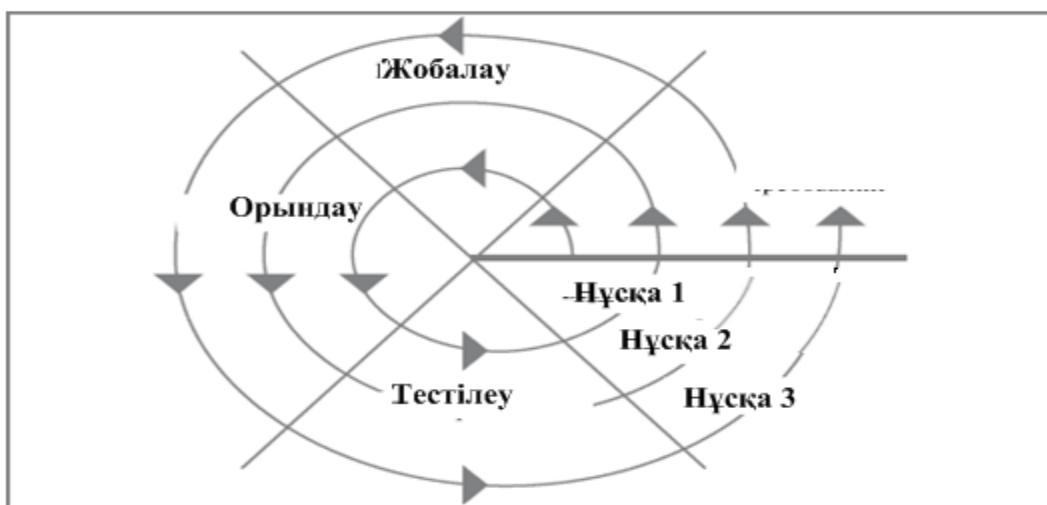
Әзірлеудің бастапқы кезеңдеріне ерекше көңіл бөлінеді –сол немесе басқа техникалық шешімдердің іске асырылуы тексерілетін және түптұлғаларды (макеттеу) құру көмегімен негізделетін стратегия жасау, талдау және жобалау. Шиыршықтың әрбір орамы өнімнің қандай да бір нұсқасын немесе оның қандай да бір құрамдасын құруды болжайды, мұнда жобаның сипаттамасы мен мақсаттары нақтыланады, оның сапасы анықталады және шиыршықтың келесі орамының жұмысы жоспарланады (Сурет 2.4).



Сурет 2.2 – Каскадты модель



Сурет 2.3 – Аралық бақылаумен жүретін кезең бойынша үлгісі



Сурет 2.4 – Спиральды модель

Тіршілік циклының ең көп таралған екі түрі қолданылады:

- каскадты модель (мерзім 1970–1985 жылдардан бастап);
- спиральды модель (мерзім 1986 жылдан бастап).

Ақпараттық жүйелердің тіршілік циклы – осы жүйеге қажеттілік туғаннан бастап оның қолданушылардың қолданысынан толық шығу уақытымен аяқталатын әртүрлі жағдайларды қамтитын олардың құрылуы және пайдалану кезеңі. Ақпараттық жүйелердің өміршеңдік кезеңі өз ішіне төрт сатыны алады: жоба алды, жобалық, енгізу, функциялану. Жобалық жұмыстардың сапасына жүйенің функциялану тиімділігі байланысты болады, сондықтан әрбір саты бірқатар кезеңдерге бөлінеді және жұмыс нәтижелерін көрсететін құжатнамалары құруды қарастырады. Жоба алды сатысында келесі кезеңдерді бөлуге болады:

- Жобалау үшін материалдарды жинау – жүйенің тұжырымдамасы нұсқасын әзірлеу және таңдау, объекті мен басқару қызметінің барлық сипаттамаларын анықтау, ішкі және сыртқы ақпараттық байланыстар ағынын, міндеттер мен жаңа технологиялық жаңдайларда жұмыс істейтін мамандар құрамын, олардың жүйенің болашақ пайдаланушысы ретіндегі дайындық

деңгейін қарастырады.

– Материалдарды талдау және құжатнаманы құру – техника-экономикалық негіздеуді жобалау, нығайтуға тапсырмалар құру. Басқарушы ақпараттық жүйені сәтті құру үшін мекеме ішінде де, сыртқы ортада да ақпараттық ағындардың өту жолдары әр тарапты зерттеледі.

Жобалау сатысы бөлінеді:

– Техникалық жобалау кезеңі – ақпараттық жүйені қамтамасыз ету және функциялық бөліктері, өндірістік, тұрмыстық, финанстық жағдайды модельдеу бойынша жобалық шешімдер құрылады, міндеттер қойлады және блок-схемалар мен олардың шешімі жүзеге асырылады.

Жұмысты жобалау кезеңі – жүйені әзірлеу және жетілдіру, құрылымды түзеу, әртүрлі құжатнамаларды түзу жүзеге асырылады: техникалық құралдарды қою, орнату, пайдалану бойынша нұсқаулықтар, лауазымдық нұсқаулықтар.

2.1.3 Ақпараттық жүйені енгізу сатысы

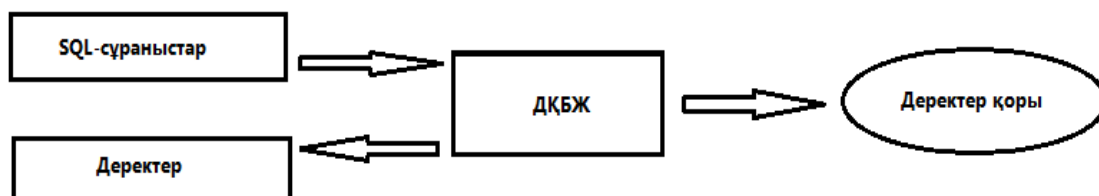
Пайдалануға енгізуге дайындық – бұл кезеңде техникалық құралдарды орнату, жүйені баптау, қызметкерлерді үйрету, сынамалы қолдану жүргізіледі. Іске қосудан бұрын жүйенің барлық құрамдастарының тәжірибелік сынақтарын жүргізу. Жұмыстарды тапсыру-қабылдау актімен рәсімделетін өнеркәсіптік пайдалануға тапсыру.

Ақпараттық жүйенің функциялану сатысында жұмыс режимінде функциялар мен басқарушы параметрлерді түзетуге болады. Сонымен қатар оперативті қызмет көрсету және басқару жүзеге асырылады. Ұйымды басқарудың ақпараттық жүйесін құру – едәуір күрделі әрі бейнетті үрдіс. Компанияның өзгертудің ең типті және қарапайым формасы автоматтандыру болып табылады. Ұйымды өзгертудің ең терек формасы – автоматтандырудан өз дамуын алған – бұл тәртіптерді рационалдау. Компанияның едәуір терең өзгеруі бизнес-процестердің реинжинирингі болып табылады. Оның мәні бизнес-процестерді талдау, ықшамдау және жаңашаландыру болып табылады. Жаңа ақпараттық жүйелер компанияның функциялануын өзгеріп немесе тіпті ондағы қызметтердің бағытын өзгерте отырып, бүкіл ұйымның құрылымын түбегейлі өзгертуі мүмкін. Компанияның қызметін өзгертудің мұндай радикалды формасы парадигманың ауысуы деп аталады. Парадигманың ауысуы жеке тәртіптер мен процестердің емес, компанияның қызметінің сипатын қайта қарауды көздейді.

2.2 Деректер қорын басқару жүйесі

2.2.1 Деректер қорын басқару жүйесі

Дерекқорды басқару жүйесі (ДҚБЖ) – дерекқорларды құруға, оларды өзекті жағдайда қуаттауға және оларда қажетті ақпараттарды іздеуді ұйымдастыруға қажетті бағдарламалық және тілдік құралдардың кешені. ДҚБЖ компьютердің сыртқы жадында күрделі құрылымды деректердің үлкен көлемін сенімді сақтау және оларға тиімді жетімділікті қамтамасыз етеді (Сурет 2.5).



Сурет 2.5 – ДҚБЖ

ДҚБЖ ге қойылатын негізгі талаптар

Деректердің қарама-қайшылықсыздығы. Қоймада жоқ тауарға тапсырыс беріліп немесе енгізу қателігінің нәтижесінде тапсырыстағы сатып алушы туралы ақпарат сатып алушылардың картотекасының деректеріне сай келмейтін жағдай болмауы тиіс. Мұндай талап бүтіндік_талабы деп аталады. Дерекқорлардың бүтіндігі ақпараттың зат саласын толық, қайшылықсыз және адекватты бейнелейтін қуаттауды көздейді. Деректердің бүтіндігіне қойылатын талаппен транзакция түсінігі байланысты. Транзакция – бір бүтін ретінде қарастырылатын дерекқорлар үстінде операциялар тізбегі.

Мысалы, жүйеде белгілі бір тауарға тапсырысты рәсімдегенде мынадай операцияларды орындау қажет: тапсырысты тіркеу және тауардың белгілі бір санын резервтеу, сонымен қатар осы тауардың қоймада азаюы. Егер деректер өзгеруінің кез келген кезеңінде шалыс болса, онда ДҚ (дерекқор) бүтіндігі бұзылады. Осындай бұзылыстардың алдын алу үшін «Тапсырысты рәсімдеу» транзакциясы енгізіледі, онда ДҚ үстінен не барлық қажетті операциялар жасалуы тиіс, не болмаса бастапқы жағдайға келтіру қажет (тауар сатылмады, оның қоймадағы саны өзгермеді).

Сақталып тұрған деректердің өзектілігі. Кез келген уақытта ДҚ дағы ақпарат заманауи болуы тиіс.

Деректерді көп аспектілі қолдану – әртүрлі көздерден жалғыз ДҚ ға ақпараттың түсуі және жетімділік ережелері мен функцияларға сай оны мекеменің кез келген бөлімінде қолдану мүмкіндігі.

Жүйені түрлендіру мүмкіндігі – оны кеңейту және деректерді түрлендіру мүмкіндігі, сонымен қатар бүтіндей жүйе үшін нұқсансыз жаңа функцияларды толықтыру.

Сенімділік – ДҚ бүтіндігі техникалық шалыстарда бұзылмауы керек.

Жетімділік жылдамдығы – талап етілген ақпаратқа жылдам жетімділікті қамтамасыз ету.

2.2.2 Дерекқордың жіктелуі

Сақталатын ақпараттың сипаты бойынша:

- фактографиялық;
- құжаттық.

Фактографиялық дерекқорда дәйектер – шынайы әлемнің объектілері қасиеттерінің (атрибууттары) нақты мәндері тіркеледі. Ақпарат белгілі бір форматта енгізіледі, айқын құрылымға ие.

Құжаттық ДҚ – барлық мүмкін болған құжаттар. Сондай-ақ бұл тек қана мәтіндік құжаттар ғана емес, сонымен қатар графика, видео және дыбыс (мультимедиа) болуы мүмкін. Мысалы: архивтер. *Құжаттық* дерекқор құрылымсыз объектілер, қамдалған тақырыптар немесе іздеудің басқаша формаланған аппаратының жиынтығын құрайды.

Деректерді сақтау тәсілі бойынша (өңдеу технологиясы бойынша):

- орталықтандырылған;
- үлестірілген (мысалы Internet).

Орталықтандырылған дерекқор бір есептеуіш жүйенің жадында сақталады. Егер бұл есептеуіш жүйе ЭЕМ желісінің құрамдасы болып табылады, мұндай базаға үлестірілген жетімділік болуы мүмкін. Дерекқорды пайдаланудың мұндай тәсілін ДҚ жергілік желіде жиі қолданады.

Үлестірілген дерекқор бірнеше, мүмкін қиылысатын немесе тіпті бір-бірін қайталайтын, есептеуіш желінің әртүрлі ЭЕМ да сақталатын бөліктерден тұрады. Осындай базамен жұмыс үлестірілген дерекқорды басқару жүйесінің (ҮДБЖ) көмегімен жүзеге асырылады.

Желілік жетімділікпен орталықтандырылған дерекқор жүйелері мынадай жүйелердің әртүрлі архитектурасын болжайды.

Файл-сервер

ДҚ жүйелерінің архитектурасы желілік жетімділікпен желінің машиналарынан бірі орталық ретінде бөлінуді болжайды (файлдар сервері). Мұндай машинада желілер жұмысшы станция функциясын атқарады, олардың көмегімен қолданушы жүйенің орталықтандырылған дерекқорға жетімділігі қуатталады. Дерекқор файлдары қолданушылар сұранысына сай олардың өңделуі жүретін жұмысшы станцияға беріледі. Сол деректерде жетімділіктің үлкен қарқындылығында ақпараттық жүйенің өнімділігі төмендейді. Қолданушылар олар жұмысшы станцияларда монополиялық қолданатын жергілікті ДҚ ды да құруы мүмкін.

Клиент-сервер

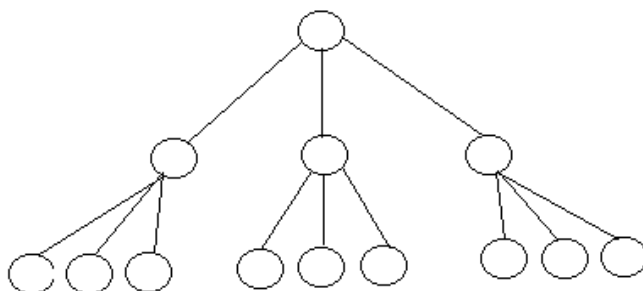
Бұл тұжырымдамада орталықтандырылған деректерқордан басқа орталық машина (дерекқор сервері) деректердің өңдеудің негізгі көлемінің орындалуын қамтамасыз ету керектігі түсініледі. Клиент беретін деректерге сұраныс (жұмысшы станциямен) серверде іздеу және деректерді шығаруды туындатады. Шығарылған деректер (бірақ файлдар емес) желі бойынша серверден клиентке тасымалданады.

2.2.3 Деректер моделі

Деректерді ұйымдастыру құрылымы бойынша:

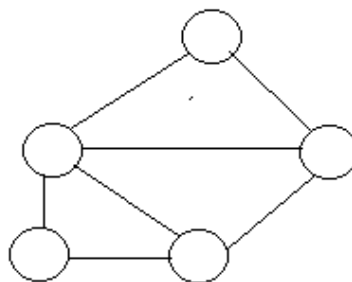
- иерархиялық;
- желілік;
- реляционды;
- объективті дерекқор;
- объективті-реляционды.

Иерархиялық ДҚ – бұл мұндай ДҚ да жазбада элементтердің реттілігі болады, бір элемент негізгі болып есептеледі, қалғандары – бағыныңқы. Жазбадағы деректер тепкішектің баспалдағы ретінде белгілі бір тізбекте реттелген, және деректерді іздеу баспалдақан баспалдаққа тізбекті «түсімен» ғана жүзеге асырылады. Мұндай жүйеде деректердің қандай да бір элементін іздеу бірнеше алдыңғы иерархиялық деңгейлерді тізбекті өтуге байланысты едәуір қиын болады. Иерархиялық ДҚ ды дискіде сақталатын файлдардың тізімдемесі, NORTON COMMANDER де көруге қол жетімді тізімдемелер ағашы, – мұндай ДҚ құрылымын көрнекі көрсету және ондағы қажетті элементті іздеу (MS-DOS жүйесімен жұмыс істегенде) құрайды. Осындай ДҚ түрлік шежіре болып табылады (Сурет 2.6).



Сурет 2.6 – Иерархиялық модельдегі байланыстар көрінісі

Желілік ДҚ – бұл дерекқор едәуір икемділігімен ерекшеленеді, өйткені онда вертикальді иерархиялық байланыстарға қосымша горизонтальді байланыстар орнату мүмкіндігі бар. Бұл деректердің талап етілген элементтерін іздеу процесін жеңілдетеді, өйткені барлық алдыңғы басқыштарды міндетті өтуді талап етпейді (Сурет 2.7).



Сурет 2.7–Желілік модельдегі байланыстар

Реляциялық дерекқор – бұл өте жиі қолданылады және әмбебап болып табылады. Деректер кесте түрінде сақталады. Бағанадағы барлық элементтер бірдей түрге (сандық, таңбалық) ие, ал әрбір бағана – қайталанбайтын ат. Кестеде бірдей жолдар жоқ. Мұндай ДҚ артықшылықтары көркемдік және деректерді ұйымдастыруың түсініктілігі, қажетті ақпаратты іздеу жылдамдығы.

Объективті дерекқор – бұл объектілік деректермен жұмыс жасау үлгісі.

Дерекқордың мұндай үлгісі көптеген жылдардан бері болуына қарамастан, жаңа болып табылады. Және оның құрылуы дерекқордың объектілік үлгісін қолданушы жеңіл қабылдағандықтан үлкен перспективалар ашады, өйткені абстракцияның оғары деңгейі ашылады.

Объективті-реляциялық ДБЖ (ОРДБЖ) – бұл объективті-бағытталған амалдарды жүзеге асыратын кейбір технологияларды қуаттайтын реляциялық ДБЖ (РДБЖ): объектілер, кластар, ұрпақ дерекқордың құрылымы мен сұраныстар тілінде іске асырылған.

2.2.4 Дерекқорды басқару жүйесінің құрылымы мен функциялары

ДҚБЖ әдетте келесі құрамдастардан тұрады:

- сыртқы және оперативтік жадта деректерді басқаруға жауап беретін ядро;

- деректерді шығару мен өзгерту және машина-тәуелсіз атқарылатын ішкі кодқа сұранысты оңтайландыруды қамтамасыз ететін дерекқор тілінің процессоры;

- ДҚБЖ мен қолданушы интерфейсін құратын деректерді қозғалтатын бағдарламаларды интерпретациялайтын атқару уақытын қуаттайтын қосалқы жүйе;

- ақпараттық жүйеге қызмет көрсету бойынша бірқатар қосымша мүмкіндіктерді қамтамасыз ететін сервистік бағдарламалар (сыртқы утилиттер).

Осыған байланысты ДҚБЖ негізгі функцияларын құруға болады.

Деректерді анықтау. ДБЖ бастапқа форма түрінде деректерді анықтау және осы анықтамаларды сәйкес объектілік формаға түрлендіру амалдарын жеткізуі керек. Яғни ДҚБЖ оларды дерекқорда сақтауға қажетті формада деректерді түрлендіреді.

Деректерді қозғалысқа келтіру. ДҚБЖ қолданушының сұрыптауға, өзгертуге немесе базадағы бар деректерді өшіруіге, немесе оған жаңа деректерді енгізуге сұраныстарын өңдеуге қабілетті болуы қажет. Яғни, ДБЖ қолданушылар мен дерекқор арасында интерфейсті қамтамасыз етеді.

Деректерді сақтау және оларға жетімділікті басқару. ДҚБЖ ЭЕМ сақтаушы құрылғысында деректерді сақтауды бағдарламалық қуаттайды және деректермен өнідірілетін барлық әрекеттерді басқарады.

Деректердің бүтіндігін қорғау және қолдау. ДҚБЖ қолданушылар сұраныстарын бақылауы және деректерді өзгеру операциясы кімге қол жетімді, ал деректерді алу операциясы кімге қол жетімді екенін анықтауы тиіс. Сонымен қатар ол ДҚ да сақталатын деректердің бүтіндігін қадағалайды. ДБЖ өзгерістерді тіреу, резервтік көшіріп алу және шалыстардан кейін дерекқордың қайта қалпына келуін жүзеге асырады.

2.2.5 Microsoft SQL server ДҚБЖ

Ірі ұйымдарда қолданылатын ең таралған ДҚБЖ ден бірі Microsoft компаниясының Microsoft SQL server ДҚБЖ болып табылады. Ол деректердің реляциялық үлгісін қолданады. Microsoft SQL server клиент-серверлік ДБЖ болып табылады, бұл оны күшті серверге ие кірі ұйымдарда қолдануға жақсы етеді.

SQL (Structured Query Language – құрылымданған сұраныстар тілі) – реляциялық дерекқор үшін дерекқорды басқару тілі. SQL өздігінен бағдарламаландырудың Тьюринг-толық тілі болып табылмайды. SQL реляциялық дерекқордағы деректерді шығару және басқарудың қарапайым стандартталған тәсілі ретінде құрылған. Қазіргі кезде SQL (басым бөлігінде Oracle ді жүзеге асыруда) дерекқорды басқару тілдерінің арасында ең танымал болып қала береді, әйтесе де бірқатар баламалары бар.

Барлық заманауи клиент-серверлік ДҚБЖ бір жалпы қасиетке ие – олармен жұмыс SQL - Structured Query Language (құрылымдалған сұраныстар тілі) тілінде жүргізіледі. Бұдан басқа, Delphi де dBase және Paradox типті қарапайым файлдық ДБЖ мен SQL көмегімен жұмыс жасау мүмкіндігінен әзірлеушілерді айырмау үшін BDE өзінің ішкі механизмдерін пайдалана отырып олар үшін ед SQL таза қолдану мүмкіндігін берді.

SQL тілінің стандартын ANSI (Америкалық стандарттардың ұлттық институты) белгілейді. Сонымен бірге, ДБЖ әрбір өндіруші тілді өте пайдалы болатын өзіндік кеңейтулерді өзінің борышы деп санайды. Кейде олар тіл стандартын біршама бұзады, әйтесе де жақсы идеялар даму үрдісіне ие және өздерінің сапасының пайдалалығынан өздігінен із жүзінде стандартқа айналады. Мұнда біз SQL ді ANSI-89 және ANSI-92 нұсқаларының толық қуатталатын барлық ДҚБЖ негізінде қарастырамыз.

SQL тілі реляциялық дерекқорда деректерді қозғалысқа келтіру, дерекқордың құрылымын анықтау және көп қолданушы ортада деректерге

жетімділік құқықтарын басқаруға арналған. Сол себепті SQL тілі 3 негізгі бөліктерден тұрады:

- деректерді қозғалысқа келтіру тілі (Data Manipulation Language, DML);
- деректерді анықтау тілі (Data Definition Language, DDL);
- деректерді басқару тілі (Data Control Language, DCL).

Осы барлық тілдер SQL тілінің бөлігі болып табылады. Олардың әр қайсысы өзінің саласына арналған SQL командаларының жиынтығынан тұрады. ДҚ мен жұмыс істеу барысында, 4 негізгі командадан тұратын деректерді қозғалысқа келтіру тілі пайдаланылады - SELECT, INSERT, UPDATE және DELETE, олардың көмегімен таңдау, қою, жаңарту және деректерді өшіру жүзеге асырылады.

Деректерді анықтау тілі ДҚ құрылымын – кестелер, индекстер және т.б. құру және өзгертуге қызмет етеді. Ол негізгі үш командадан тұрады - CREATE, ALTER және DROP (құру, өзгерту және өшіру), олардың әрқайсысы 6 объектілерден бірін – дерекқор, кесте, виртуалды кесте, индекс, триггер немесе сақталатын тәртіпті әрекетке келтіреді. Осылайша, мысалы CREATE командасы үшін бір келесі 6 нұсқаны аламыз:

- CREATE DATABASE – дерекқор құру;
- CREATE TABLE – кесте құру;
- CREATE VIEW – виртуалды кесте құру;
- CREATE INDEX – индекс құру;
- CREATE TRIGGER – триггер құру;
- CREATE PROCEDURE – сақталатын тәртіпті.

2.2.6 Кесте құру және өшіру

SQL тілінде кесте CREATE TABLE командасы арқылы қолданылады:

```
CREATE TABLE <Кесте атын жазамыз>
                (<Әріп аты> <типi>
                .
                .
                .
                <Әріп аты> <типi>);
```

Мысалы:

```
CREATE TABLE      Dictionary
                    (d_id Integer
                    english VARCHAR(15)
                    kazakh NCHAR(15);
```

Кестені өшіру үшін келесі инструкция қолданылады:

```
DROP TABLE <Кестенің аты>;
```

Мысалыға DROP TABLE <Dictionary>; Dictionary атты кестеге қатысты файлдардың бәрі өшіріледі.

Кесте өрістерінің құрамын өзгерту.

Өрістердің құрамын өзгерту деген өрістерді өшіру не қосуды білдіреді, және кестені басқа ешқандай қосымшалар қолданбауы қажет. Өрістердің құрамын өзгерту үшін ALTER TABLE инструкциясын қолданамыз:

```
ALTER TABLE <кестенің атын жазамыз>  
ADD <өріс аты> <өріс типі>,  
DROP <өріс аты>;
```

ADD кестеге жаңадан өріс қосады, ал DROP кестеден атымен көрсетілген өрісті өшіріп тастайды. DROP және ADD бір–біріне мүлдем қатысы жоқ және еркін орналаса алады.

3 Деректер қорын жобалау

3.1 ДҚ құрудың алғашқы кезеңі

Жұмыстың жалпы сипаттамасы

Бұл жобада ағылшын тілін үйренуге арналған бағдарлама және, ағылшынша-орысша-қазақша сөздік құрастыру қарастырылған. Бұл бағдарлама сөздік, арнайы сабақтар, кестелер, ағылшынша-қазақша мақал-мәтелдер, ағылшын тілінен білімін тексеруге арналған тесттерден тұрады.

Дерекқордағы шешілетін мәселелер

Ағылшынша-орысша-қазақша сөздікке дерекқор жасау және әр тілде жеке іздеу сұраныстарын жасау. Етісіктерді сұрыптау үшін қосымша кесте құру және сұраныстар жасау. Ағылшынша-қазақша мақал-мәтелдер үшін жеке кесте және тілашардың әр тақырыбына жеке кесте құру керек. Оларға сұраныстар жасау.

3.2 Концептуалды жобалау

Қажетті мәлімет қолданушылардың талабы

Жоба пайдаланушының барлық негізгі талаптарын қанағаттандыру керек. Бұл жүйенің негізгі пайдаланушылары университет мұғалімдері мен студенттері болады. Негізгі талаптар алынған ақпараттың ұсынылатын тиімділік деңгейінде негізделеді. Жүйеге қойылатын негізгі талаптардың тізімі:

- жүйе қолданыста қарапайым болу керек. интерфейс ыңғайлы болу керек;
- жүйе қауіпсіздікті қамтамасыз ету керек;
- жүйе дерекқордың тұтастығын қамтамасыз ету керек;
- жүйе толығымен интеграциялану керек, яғни мәліметтердің артылуы және артық жаңартулар болмау керек.

3.3 ER моделін құру

Деректер қорын құрудың алғашқы кезеңінде нысандар жиынтығын анықтап алу керек. Бұл соңғы пайдаланушылар мен жобалаушы көзқарасы бойынша жүйе объектілері туралы ең маңызды ақпарат болып табылады. Бағдарлама 3.1-кестеде көрсетілген нысандарды қолданады.

Кесте 3.1 – Дерекқордың нысандары

Нысан аты	Сипаттамасы
Dictionary	Ағылшынша-орысша-қазақша сөздік
Verbs	Ең көп қолданылатын етістіктер
pastsimple	Irregular verbs

Кесте 3.1 нің соңы

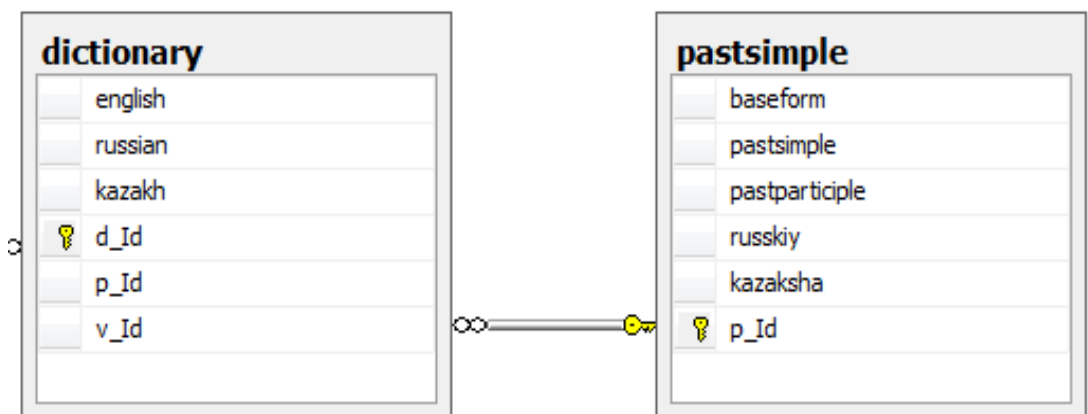
Нысан аты	Сипаттамасы
Makal_matel	Мақал-мәтелдер
Surak	Сұрақ қою
Salemdesu	Сәлемдесу

Koshtasu	Қоштасу
Konak_ui	Қонақ үйде
Sayaxat	Саяхат
Memleket	Мемлекет
Jumis	Жұмыс
Kalada	Қалада
Sauda	Сауда
Aua_rayi	Ауа-райы
Uakit	Уақыт
Tilder	Тілдер
Tanisu	Танысу
Tamaktanu	Тамақтану

Жобалаушы және соңғы қолданушы нысандарды анықтау кезінде келісімге келу қажет. Жобалаушы нысандар арасындағы байланыстарды операциялардың сипаттамасына негізделіп анықтайды. Нысандар арасындағы байланыстар операциялардың толық сипаттамасы негізінде құрылған бизнес-ережелерге негізделеді.

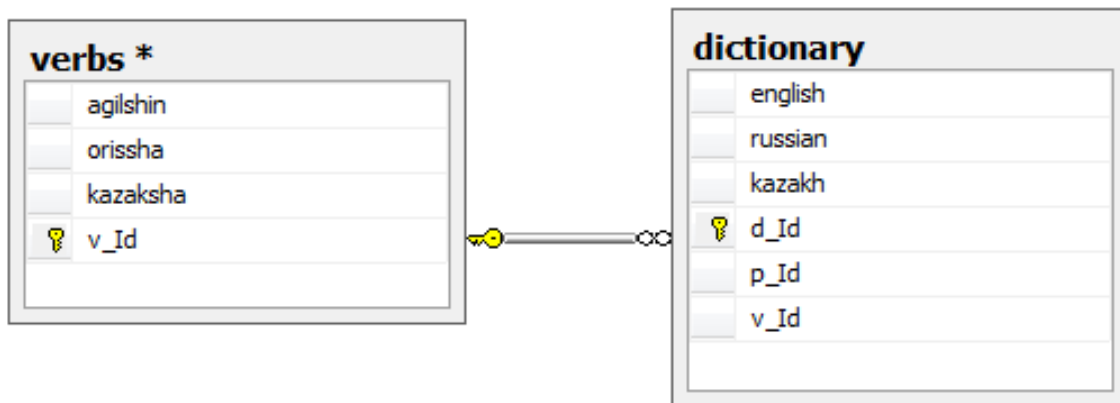
3.3.1 Бизнес ережелер

1 Кесте кодтары арқылы байланысады және сұраныстар жасалады (Сурет 3.1).



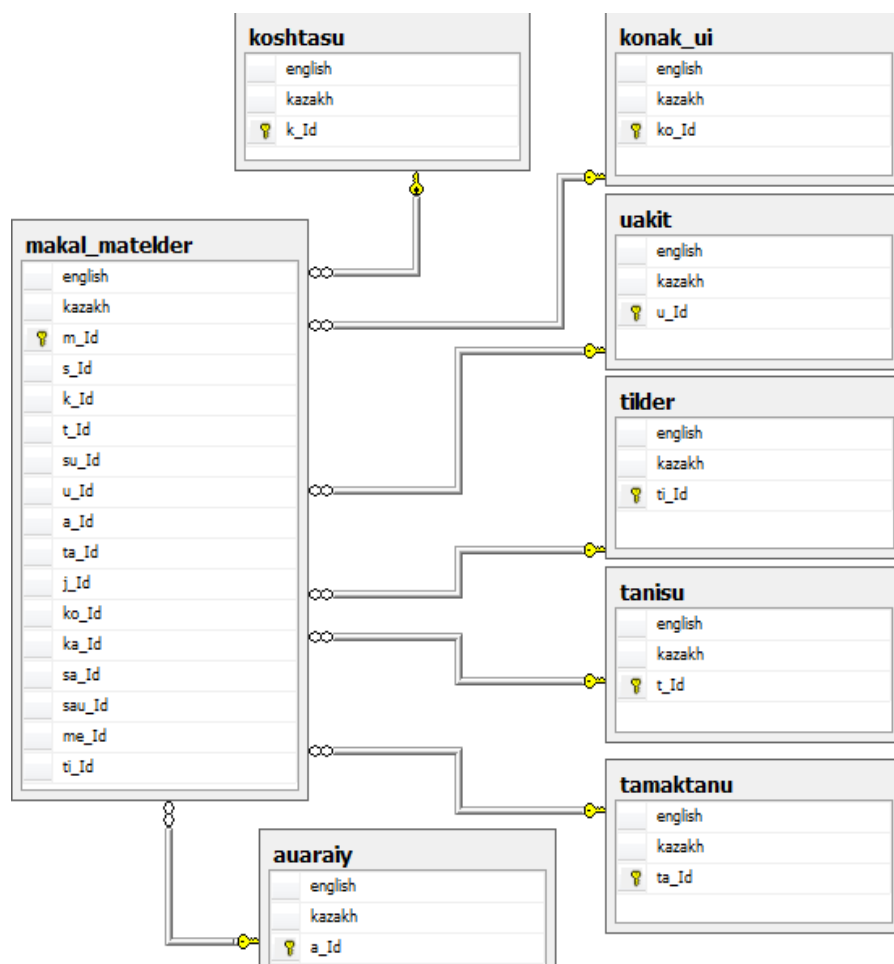
Сурет 3.1 – 1 бизнес ереже үшін ER сегменті

2 Сұрыптау жасау үшін (Сурет 3.2).



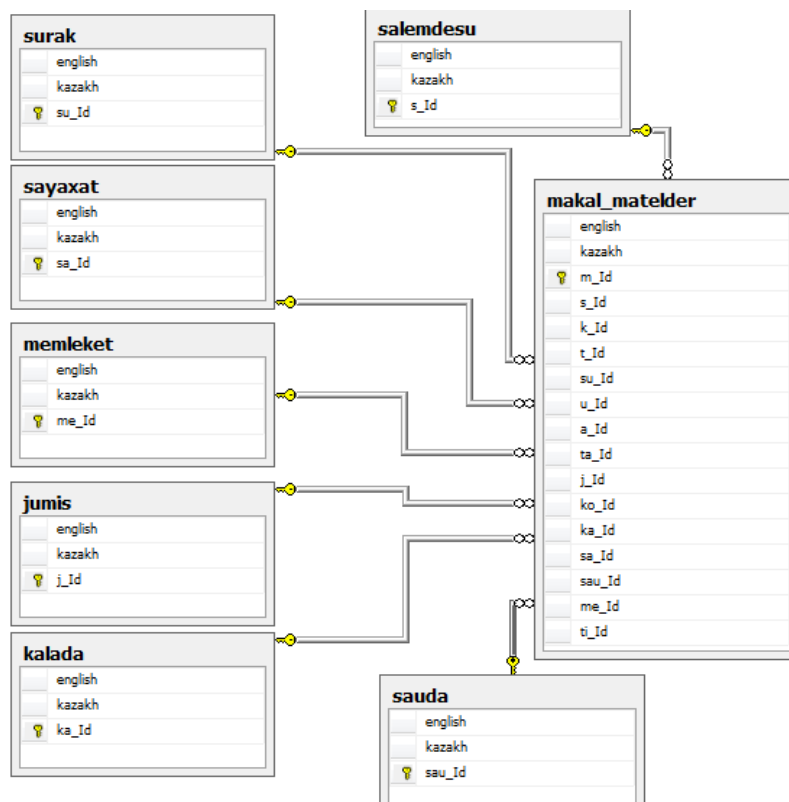
Сурет 3.2 – 2 бизнес ереже үшін ER сегменті

3 Керекті тақырыпты таңдағанда осы кестелерден сұраныс арқылы шығып тұрады (Сурет 3.3).



Сурет 3.3 – 3 бизнес ереже үшін ER сегменті

4 Керекті тақырыпты таңдағанда осы кестелерден сұраныс арқылы шығып тұрады (Сурет 3.4).



Сурет 3.4 – 4 бизнес ереже үшін ER сегменті

3.3.2 Реляционды қатынастарды құрастыру

Әр реляционды қатынастар белгілі бір нысанға сәйкес болады және оған сол нысанның барлық атрибуттары, шарттары енгізіледі. Кестелерді байланыстыру үшін әр нысанға алғашқы кілт және сыртқы кілттерді беру қажет. Қатынастар 3.2 – 3.19 – кестелерінде келтірілген. Әр қатынас үшін оның ішкі аты, типі және ұзындығы көрсетіледі. Мәліметтердің типін келесідей белгілейміз: int – сандық, nchar – символдық (Юникод)қазақ тіліндегі әріптерді оқу үшін, char – символдық.

К е с т е 3.2 – Dictionary қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Ағылшынша	english	varchar(17)	Сыртқы кілт
Орысша	russian	varchar(17)	Қажетті өріс
Қазақша	kazakh	nchar(17)	Сыртқы кілт
Сөздік коды	d_Id	int	Алғашқы кілт
Етістік коды	p_Id	int	Сыртқы кілт

К е с т е 3.3 – Verbs қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Ағылшынша	agilshin	varchar(20)	Қажетті өріс

Орысша	orissha	varchar(20)	Қажетті өріс
Қазақша	kazaksha	nchar(20)	Қажетті өріс
Етістік коды	v_Id	int	Алғашқы кілт

К е с т е 3.4 – Pastsimple қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Негізгі етістік	baseform	varchar(18)	Қажетті өріс
Өткен шақта	pastsimple	varchar(18)	Қажетті өріс
Есімше түрінде	pastparticiple	varchar(18)	Қажетті өріс
Орысша	russkiy	varchar(18)	Қажетті өріс
Қазақша	kazaksha	nchar(18)	Қажетті өріс
Етістік2 коды	p_Id	Int	Алғашқы кілт

К е с т е 3.5 – Surak қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сұрақ қою	su_Id	int	Алғашқы кілт
Ағылшынша	English	varchar(70)	Қажетті өріс
Қазақша	Kazakh	nchar(70)	Қажетті өріс

К е с т е 3.6 – Salemdesu қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сәлемдесу	s_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.7 – Makal-matelder қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Мақал-мәтел коды	m_id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс
Сұрақ қою	su_Id	int	Сыртқы кілт
Сәлемдесу	s_Id	int	Сыртқы кілт
Қоштасу	k_Id	int	Сыртқы кілт
Қонақ үйде	ko_Id	int	Сыртқы кілт
Саяхат	sa_Id	int	Сыртқы кілт

Кесте 3.7 – нің соңы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Мемлекет	me_Id	int	Сыртқы кілт
Жұмыс	j_Id	int	Сыртқы кілт

Қалада	ka_Id	int	Сыртқы кілт
Сауда	sau_Id	int	Сыртқы кілт
Ауа-райы	a_Id	int	Сыртқы кілт
Уақыт	u_Id	int	Сыртқы кілт
Тілдер	ti_Id	int	Сыртқы кілт
Танысу	t_Id	int	Сыртқы кілт
Тамақтану	ta_Id	int	Сыртқы кілт

К е с т е 3.8 – Koshtasu қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Қоштасу	k_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.9 – Konak_ui қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Қонақ үйде	ko_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.10 – Sayaxat қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Саяхат	sa_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.11 – Memleket қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Мемлекет	me_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.12 – Jumis қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Жұмыс	j_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс

Кесте 3.12 – нің соңы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.13 – Sauda қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сауда	sau_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.14 – Kalada қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Қалада	ka_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.15 – Ауа_райі қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Ауа-райы	a_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.16 – Uakit қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Уақыт	u_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.17 – Tilder қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тілдер	ti_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.18 – Tanisu қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Танысу	t_Id	int	Алғашқы кілт
Ағылшынша	english	varchar(70)	Қажетті өріс
Қазақша	kazakh	nchar(70)	Қажетті өріс

К е с т е 3.19 – Тамақтану қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тамақтану	ta_Id	int	Алғашқы кілт
Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Ағылшынша	english	varchar(70)	Қажетті өріс

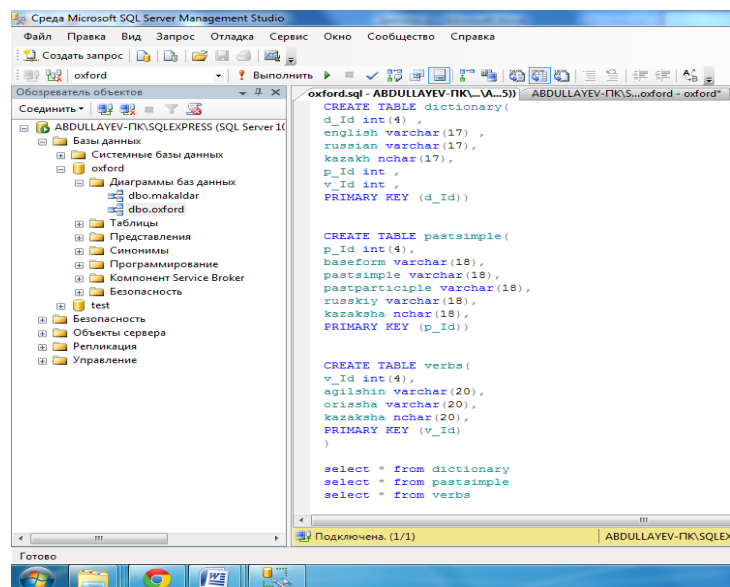
Қазақша	kazakh	nchar(70)	Қажетті өріс
---------	--------	-----------	--------------

3.3.3 Логикалық жобалау

Логикалық (даталогикалық) жобалау – нақты деректер үлгісі негізінде дерекқорлар схемасын құру, мысалы, деректердің реляциялық үлгісі. Деректердің реляциялық үлгілері үшін даталогиялық үлгі – әдетте біріншілік кілттерді көрсетумен, қатынастар, сонымен қатар ішкі кілттен тұратын қатынастар арасында «байланыстар» схемасының жиынтығы. Тұжырымдамалық үлгіні логикалық үлгіге түрлендіру, ережеге сай, формалық ережелер бойынша жүзеге асырылады. Бұл кезең едәуір дәрежеде автоматтандырылған болуы мүмкін. Логикалық жобалау кезеңінде деректердің нақты үлгісінің ерекшелігі ескеріледі, бірақ нақты ДБЖ ерекшелігі ескерілмеуі мүмкін.

Кестелер құру

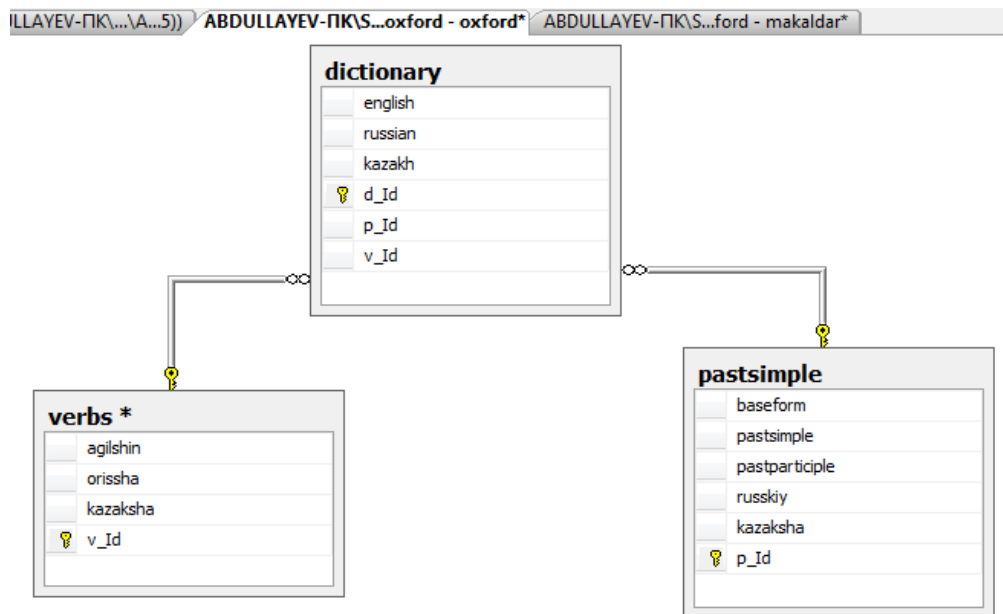
Кестелерді CREATE TABLE командасымен құруға болады. CREATE TABLE бағандар атын және кесте атын анықтайды және ол деректердің типін және бағандардың өлшемін анықтайды. Ең ыңғайлысы бұл генерацияланған скриптті Query Analyzer-де (MS SQL Server) қосу. Кесте құруға арналған скрипт А қосымшада көрсетілген (Сурет 3.8).



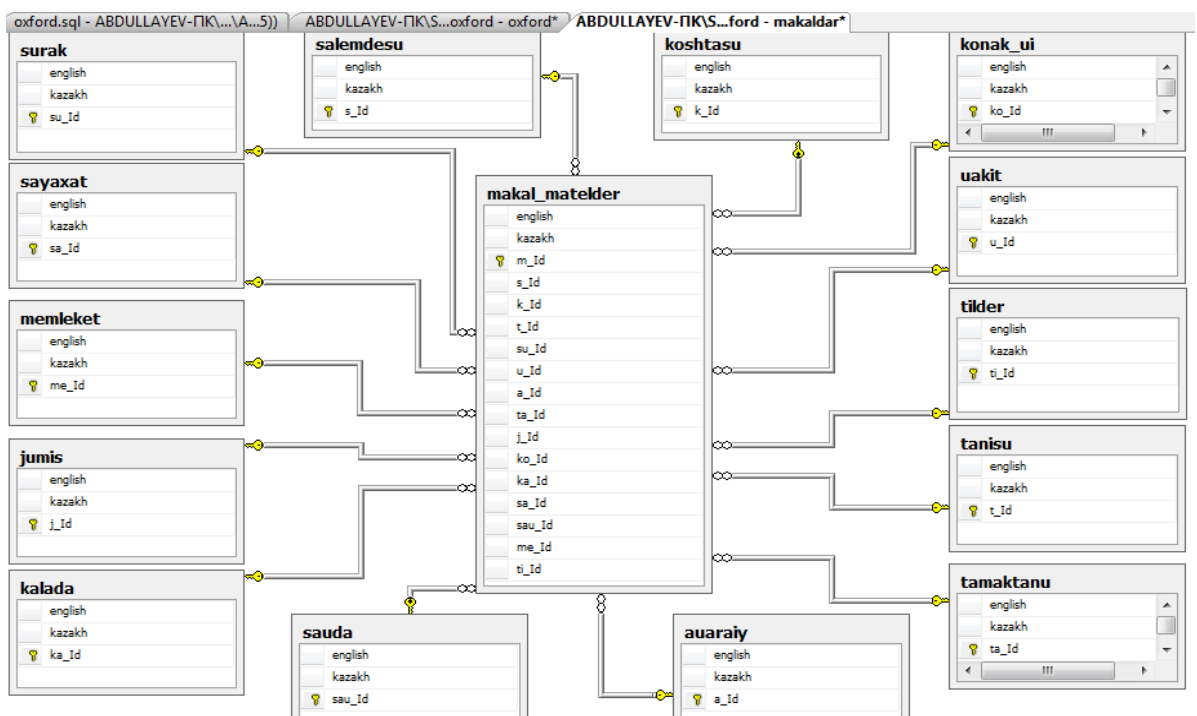
Сурет 3.8 - Query Analyzer-де кесте құру

ДҚ диаграммасы

Концептуалды моделді дерекқордың сәйкес құрылымына енгізілу тәсілі қандай болса да, ДҚ реляционды сұлбасы жобаға сәйкес болу керек (Сурет 3.9–3.10).



Сурет 3.9 – ДҚ диаграммасы



Сурет 3.10 – ДҚ диаграммасы

3.3.4 Көрсетілімдер және сұраныстар

Көрсетілімдер бұл кестелер, бірақ оның құрамы басқа кестелерден алынады. Олар сұраныстарда дәл негізгі кестелер сияқты жұмыс істейді, бірақ оларда өзінің жеке деректері жоқ.

```
Сөздікте үш тілде іздеу жасауға
SQL.Add('select * from dictionary where english like ');
SQL.Add('select * from dictionary where russian like ');
SQL.Add('select * from dictionary where kazakh like ');
Сұрыптама жасауға
'select english, russian, kazakh from dictionary'
'select agilshin, orissha, kazaksha from verbs'
'select english, russian, kazakh from dictionary, pastsimple where
english=baseform'
```

Және мақалдар терезесінде тілашарды шығару үшін төмендегідей сұраныс қолданылды 14 тақырыпқа байланысты 14 сұраныс жазылды.

```
'select english, kazakh from Salemdesu'.
```

3.4 Физикалық жобалау

3.4.1 Дерек қорына қосымша құру

Бұл бағдарламада деректер қорына қосымша құру үшін Embarcadero RAD Studio 2010 бағдарламалық құралы және дизайн жасауда Adobe Photoshop бағдарламалық құралы қолланылды.

Embarcadero RAD Studio 2010 – Windows, .NET қосымшаларын, веб-қосымшаларды және дерекқорлар қосымшаларын жылдам дайындаудың кешенді шешімі болып табылады. Ол өзіне Windows тің әртүрлі платформаларына және дерекқорға 5 есе жылдам қосымшаларды құруға мүмкіндік бере отырып, өз ішіне Delphi®, C++Builder® және Delphi®Prism™ ді алады. RAD Studio құралдары компьютерлер, жұмыс станциялары, сенсорлық дисплейлер ақпараттық терминалдар мен Интернет үшін қосымшалардың құрылуын жылдамдатады, сондықтан оларды 2 миллионға дейін әзірлеушілер пайдаланады. Осы амалдардың құрамына жылдам компиляторлар, қолданушы интерфейсімен қосымшаларды жасау үшін визуалды орта, визуалды компоненттердің платформасы (VCL), сонымен қатар дерекқорлар мен қызметке қосылу үшін кең мүмкіндіктер кіреді. Әріптестер мен әзірлеушілер бірлестігі әрбір нақты қосымша, технология, функция және міндеттер үшін мыңдаған қосымша компоненттерді ұсынады. Басқа бірде-бір шешім осындай платформа, деректер көзі, БЖ (бағдарламалық жасақтама) типі, архитектура және қолданулардың осындай саны үшін сапалы, жоғары өнімді қосымшаларды жасауда қажетті функциялық, икемділік және басқарушылыққа ие емес.

Adobe Photoshop – Adobe Systems фирмасында әзірленген және таралған көп функциялы графикалық редактор. Негізінен растрленетін бейнелермен жұмыс істейді, алайда кейбір веткорлық құралдарға ие. Өнім растрлік

бейнелерді редакциялаудың коммерциялық амалдарында нарықта жетекші, және Adobe фирмасының ең танымал өнімі болып табылады. Қазіргі кезде Photoshop Mac, Windows платформаларда, iOS және Android модильді жүйелерінде қол жетімді

Embarcadero RAD Studio Delphi 2010 бағдарламасынды мәліметтер қорымен жұмыс істеу қосымшасы мәліметтерге енуді қамтамасыз ететін компоненттерден, өрістердің мәліметтерін карап шығу және редакциялау мүмкіндіктерінен тұрады. Мәліметтерді бейнелеу компоненті – Data Controls қосымшасында болады. DataSource компоненті болса мәліметтерді редакциялау. DataSource мәліметтер көзін тез таңдауды, сол компоненттердің қолданып, мысалы DBGrid компонентінде кестеден мәліметтерді көру үшін немесе SQL-сұраныстарды жазуға жіне нәтижесін көруге мүмкіндік береді.

3.4.2 Бағдарламаны сипаттау

Бағдарламаның алғашқы беті

Бағдарламаның алғашқы беті ағылшын тілі грамматикасы, тест тапсыру, Oxford 3000 сөздігі, ең көп қолданылатын етістіктер, irregular verbs етістіктері, ағылшынша-қазақша мақал-мәтелдерден тұрады (Сурет 3.10).



Сурет 3.10 –Бағдарламаның алғашқы терезесі



Сурет 3.11 –Ағылшын тілі грамматикасы терезесі

Form27

THE INFINITIVE ТҰЙЫҚ ЕТІСТІК (ИНФИНИТИВ)

Жақсыз етістіктердің бірі – The Infinitive (Тұйық етістік). Тұйық етістік – етістіктің ерекше түрі. Тұйық етістік етістіктің негізгі түбірі. Тұйық етістікке істеді? Қайтті? Деген сұрақтарға жауап береді. Олар: to go, to do, to get, to ask, to meet, to eat, to read, т.б.

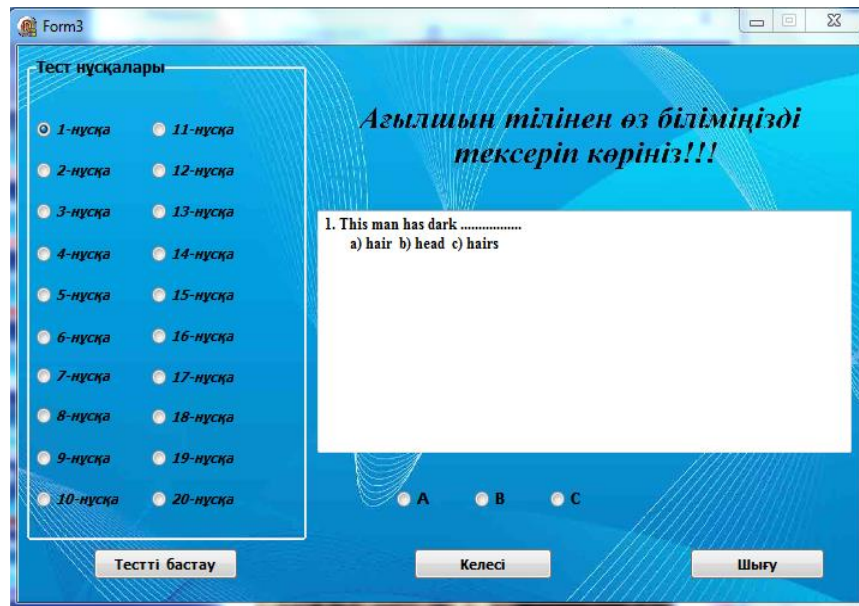
1. Тұйық етістік сөйлемде бастауыш, баяндауыш, толықтауыш, анықтауыш, пысықтауыштың, қызметін атқарады.
2. Тұйық етістік етістіктің де белгілерін білдіреді:
 - а) Тұйық етістік тура толықтауыштың қызметін атқарады:
I told him to post the letter. – Мен хатты жіберу туралы онымен сөйлестім.
 - ә) Тұйық етістік үстеумен анықталады:
I asked him to speak slowly. – Мен одан ақырын сөйлеуін өтіндім.
 - б) Тұйық етістікке тән шақ және етіс категориясы бар. Ағылшын тілінде сабақты етістіктер төрт формада негізгі етісте және екі формада ырықсыз етісте беріледі.

(Тұйық етістіктің сөйлемдегі қызметі)	
Subject (Бастауыш)	
To skate is pleasant.	Қоғалыммен сырғанау жақсы.
To walk in the garden was a pleasure.	Бақшада серуен жасау жақсы болатын.
Nominal predicate (Есімді байланыс)	
Your duty was to inform me about it immediately.	Сіздің міндетіңіз бұл туралы маған бірден хабарлау болатын.
This is for you to decide.	Бұны өзіңіз шешуіңіз керек.
Our plan is to go to the Saraygash for the summer.	Біздің жоспарымыз – жазда Сарыағашқа бару.
Verbal predicate (Етістікті байланыс)	
She began to translate the article.	Ол мақаланы аудару бастады.
He is ready to go there.	Ол онда баруға дайын.
He can speak English.	Ол ағылшынша сөйлей алады.
Object (Толықтауыш)	
He asked me to wait a little.	Ол менен аз күте тұрамын деп сұрады.
I told him to go there.	Мен оған онда бар деп айттым.
Attribute (Анықтауыш)	
He expressed a desire to help me.	Ол менен етіш көмек сұрады.
We have no intention to order these goods.	Біз бұл тауарға тапсырыс жасауға ниетіміз жоқ.

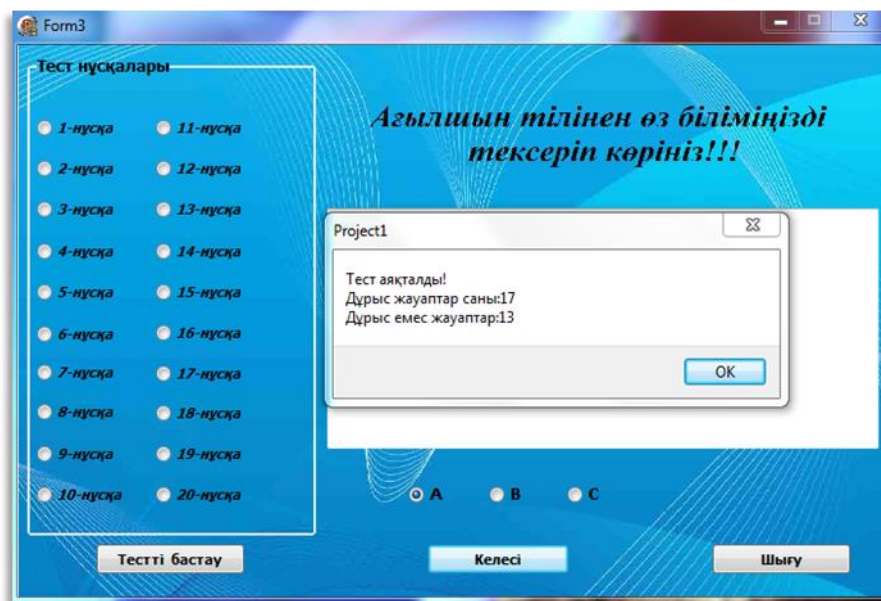
Сурет 3.12 – Сабақтар

Тест тапсыру бөлімі

Тест тапсыру бөлімінде ағылшын тілінен білім деңгейін тексеруге арналған 20 нұсқадан тұратын тесттер берілген. Тест біткенде дұрыс және дұрыс емес жауаптар санын шығарып береді (Сурет 3.13-3.14).



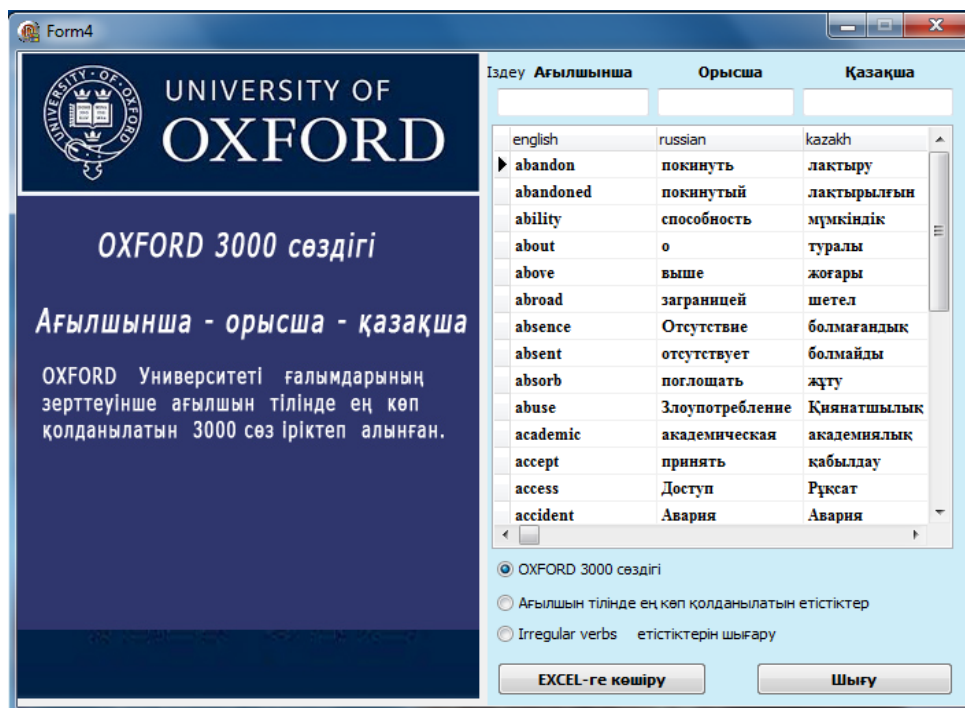
Сурет 3.13 – Тест тапсыру терезесі



Сурет 3.14 –Тест жауаптарын шығару

Oxford 3000 сөздігі

Oxford 3000 сөздігі Оксфорд университеті ғалымдары зерттеген ең көп қолданылатын сөздерден тұрады. Сөздікте ағылшынша, орысша, қазақша іздеуге болады. Және ең көп қолданылатын етістіктерді сұрыптауға болады (Сурет 3.15 - 3.16).



Сурет 3.15 – Oxford 3000 сөздігі



Сурет 3.16 – Іздеу

Excel-ге көшіру

Кез келгенін таңдап Excel-ге көшіріп алуға болады (Сурет 3.16).

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	english	russian	kazakh									
3	abandon	покинуть	лақтыру									
4	abandone	покинуты	лақтырылғын									
5	ability	способно	мүмкіндік									
6	about	о	туралы									
7	above	выше	жоғары									
8	abroad	загранице	шетел									
9	absence	Отсутстви	болмағандық									
10	absent	отсутству	болмайды									
11	absorb	поглощат	жүту									
12	abuse	Злоупотр	Қиянатшылық									
13	academic	академич	академиялық									
14	accept	принять	қабылдау									
15	access	Доступ	Рұқсат									
16	accident	Авария	Авария									
17	by accider	случайно	кездейсоқ									

Сурет 3.16 – Excel-ге көшіру

Irregular verbs етістіктері.

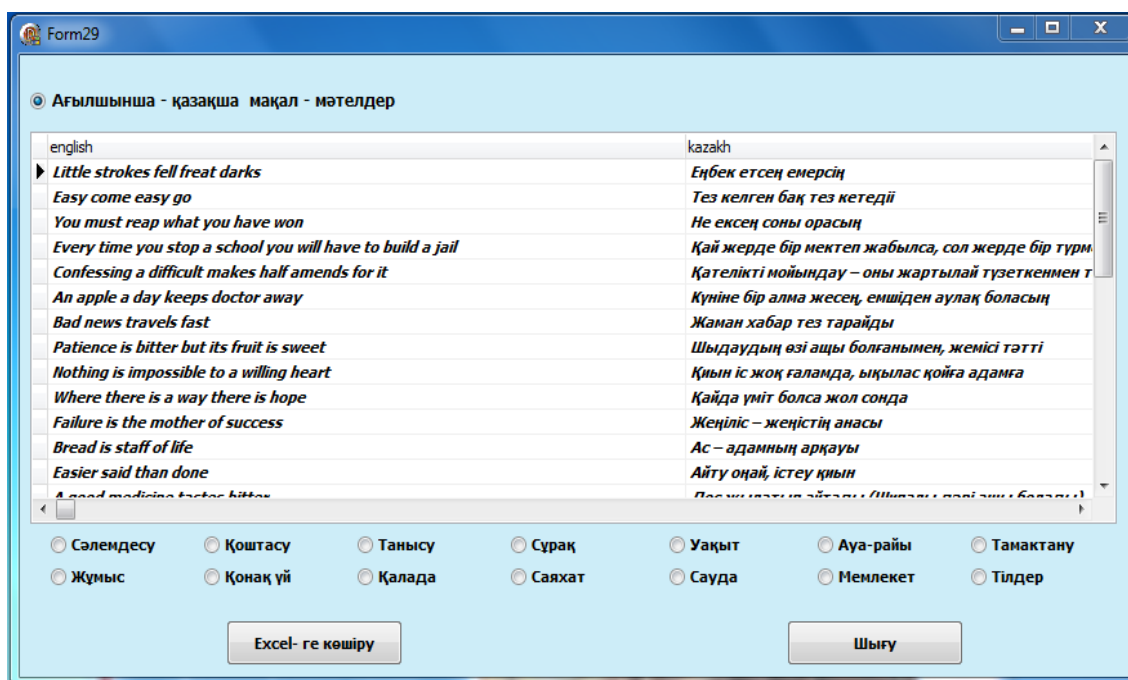
Irregular verbs ағылшын тілінде етістіктердің түрленуі, өткен шақ, есімше түріндегі. Бұл жерде де керекті етістікті жазып іздеуге және Excel-ге көшіріп алуға болады (Сурет 3.17).

Негізгі түрі	Өткен шақ	Есімше түрде	Орысша	Қазақша
baseform	pastsimple	pastparticiple	russkiy	kazaksha
▶ go	went	went	ходить	бару
eat	ate	ate	кушать	жеу
arise	arose	arisen	подняться	көтерілу
be	was; were	been	быть	болу
awake	awoke	awaked; awoke	будить	ояту
bear	bore	born; borne	родить	туылу
beat	beat	beaten	бить	ұру
become	became	become	становиться	болу
befall	befell	befallen	случиться	болду
beget	begot;	begotten	порождать	тудыру
begin	began	begun	начинать	Бастау
begird	begirt	begirt	опоясывать	буып алу
behold	beheld	beheld	зреть	туындау
bend	bent	bent;	наклоняться	еңкею
bereave	bereft; bereaved	bereft; bereaved	лишать	айрылу
beseech	besought;	besought;	упрашивать	көндіру

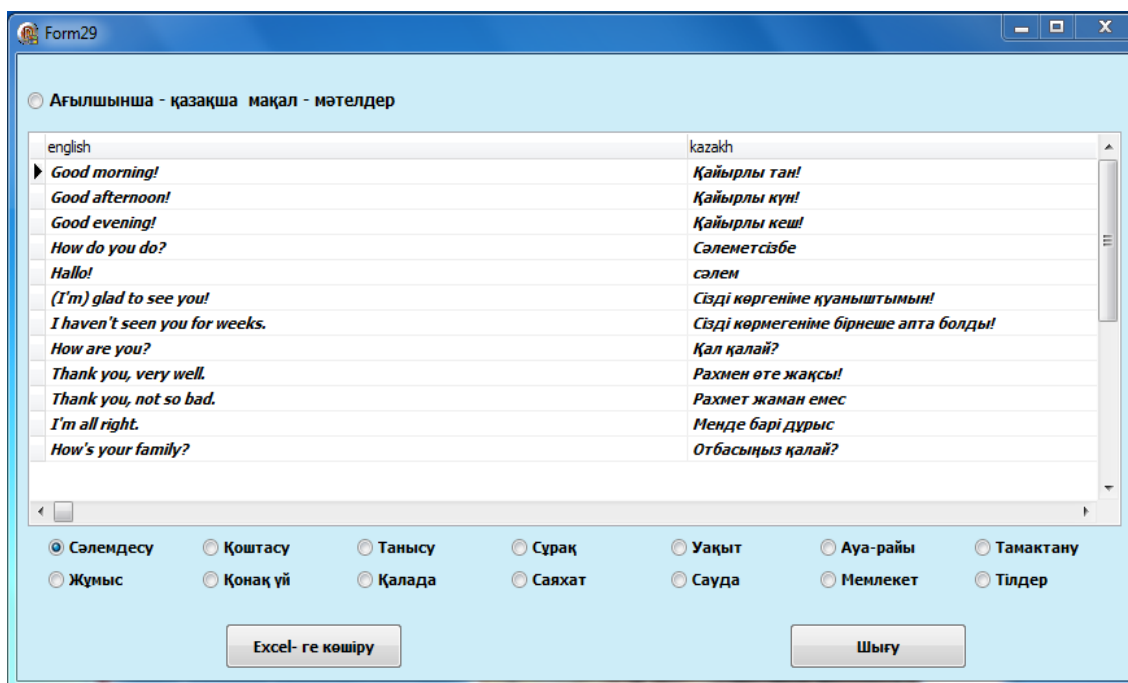
Сурет 3.17 – Етістіктің түрленуі(Irregular verbs)

Ағылшынша-қазақша мақалдар және тілашар терезесі

Ағылшынша – қазақша мақалдар аудармасы және күнделікті қолдануға арналған әр түрлі тақырыпқа қатысты тілашар бар. Керекті тақырып немесе мақалдарды таңдап шығаруға болады. Және Excel-ге көшіріп алуға болады (Сурет 3.18 – 3.19).



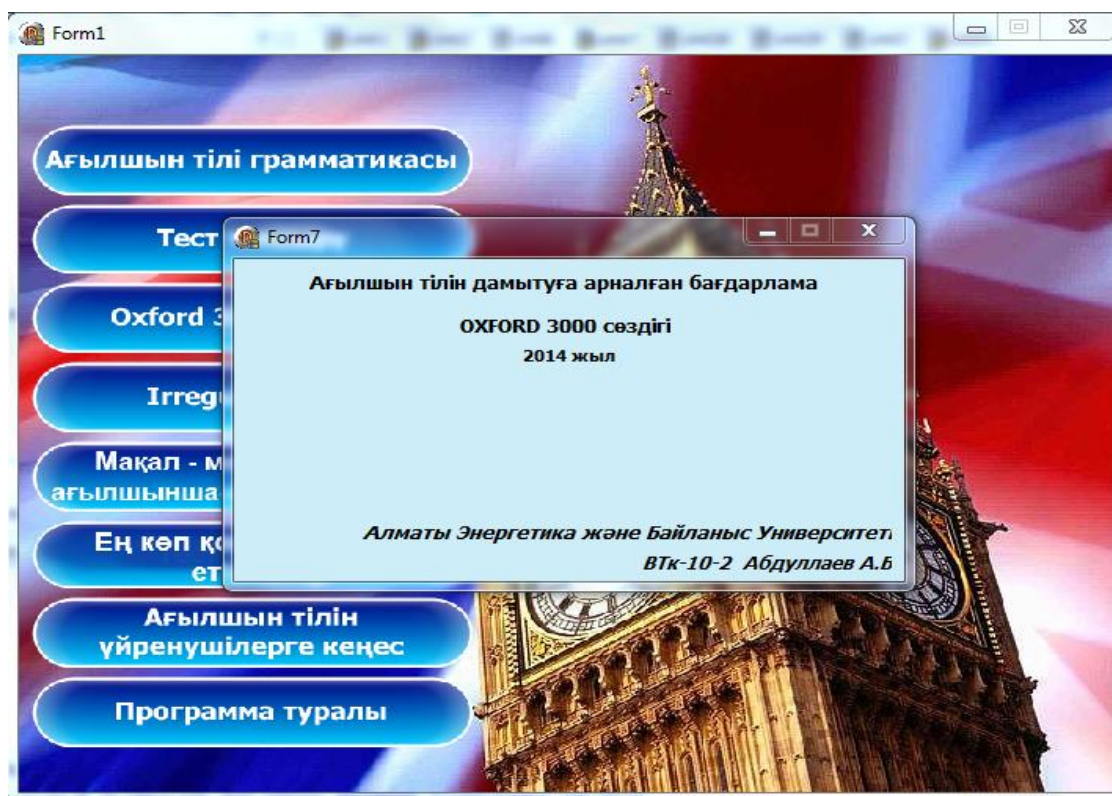
Сурет 3.18 – Ағылшынша–қазақша мақалдар



Сурет 3.19 – Тілашар терезесі

Бағдарлама туралы

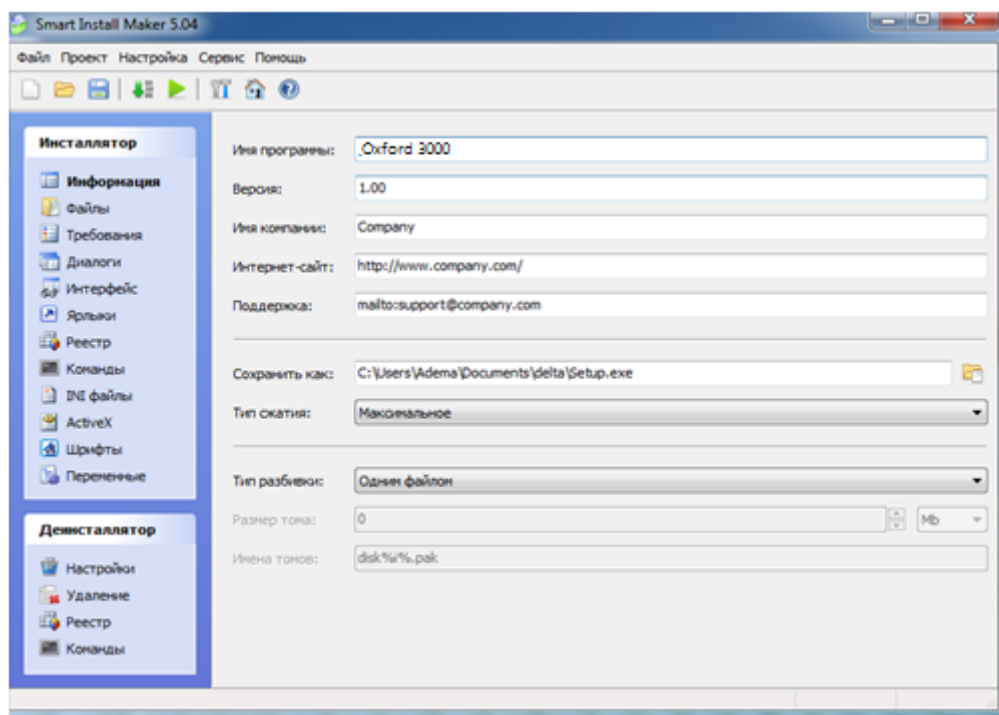
Бұл жерде бағдарлама туралы жазылған (Сурет 3.20).



Сурет 3.20 – Бағдарлама туралы терезесі

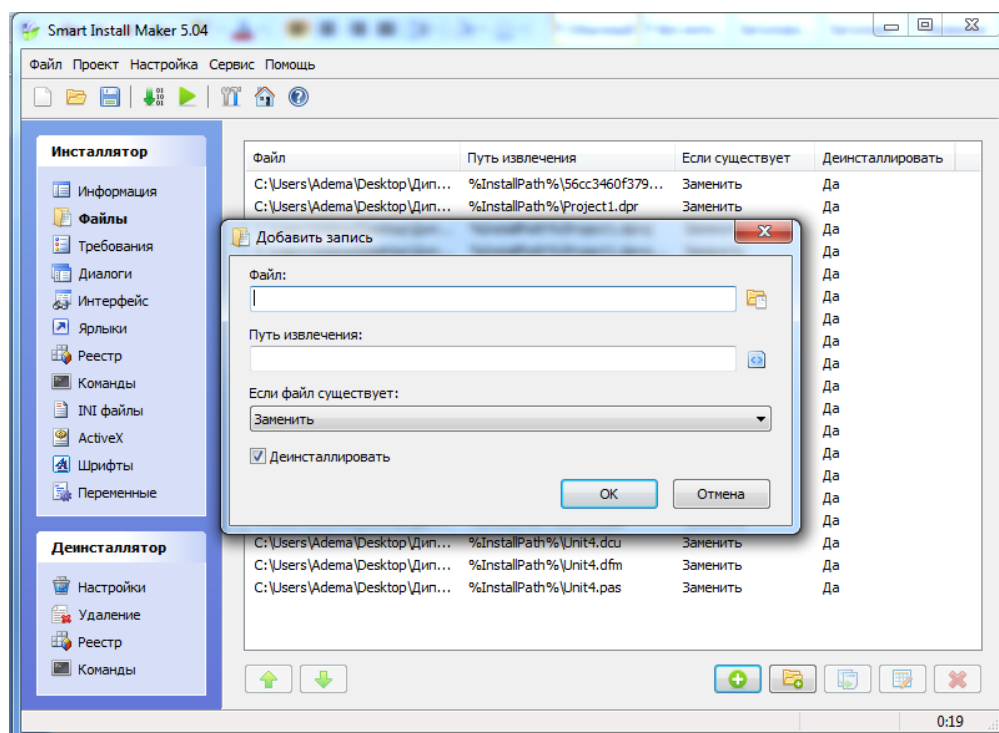
3.5 Орнату дискісін құру

Орнату дискісін құру үшін Smart Install Marker бағдарламасын қолдандым . Smart Install Maker Microsoft Windows ортасы үшін сенімді және көркем инсталляциялық пакеттерді құруда функциялық мүмкіндіктерге бай. Утилита қысу және кіші өлшеммен ықшам инсталляторларды құру үшін Cabinet мықты алгоритмін жеткізеді, көптілді (25 тілге дейін) инсталлятор құруды қуаттайды, сонымен қатар бағдарлама-орнату сыртқы түрін басқару үшін құралдарды жеткізеді. Smart Install Maker басқа мүмкіндіктерінің арасында жүйенің көрсетілген директорияларында деинсталляторлар, ярлыктарды құру, жаңа қаріптер мен операциялық жүйеде кеңейтулерді тіреу, ActiveX және айнмалыларды қуаттау, орнатылған .NET, Framework тексеру, инсталляторды орнатушы дискілерге бөлу, орнатушы файл туралы ақпаратты баптау, жүйелік талаптар, сонымен қатар оның сыртқы түрі мен т.б. бөлуге болады. (Сурет 3.21).



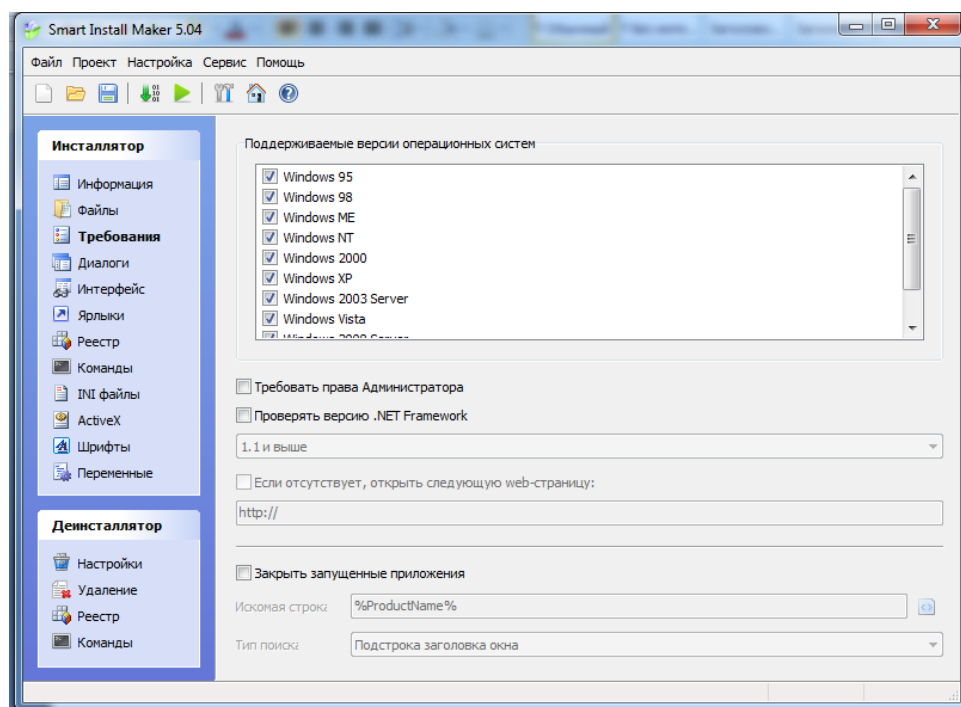
Сурет 3.21 – Бағдарламаға қатысты ақпараттар енгізу

Одан кейін «Файл» батырмасын басып, керек жазбаны тандаймыз (3.22-Сурет).



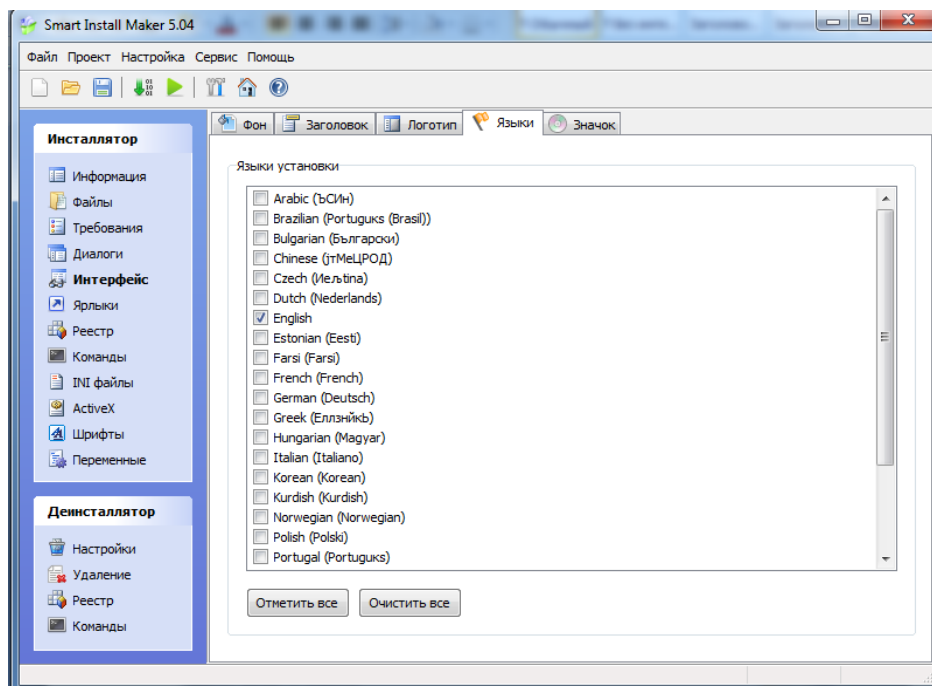
Сурет 3.22 – Жазба қосу

«Требования» батырмасын басып, керекті операциялық жүйеге орнатылу мүмкіндігін таңдаймыз (Сурет 3.23).



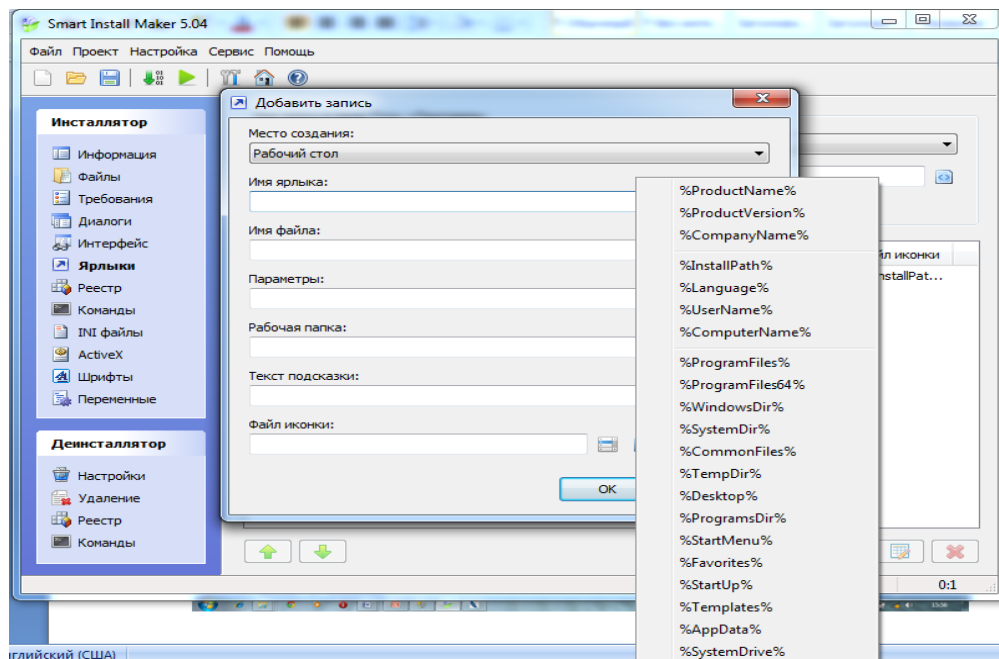
Сурет 3.23 – Операциялық жүйені таңдау

«Интерфейс» батырмасын басып, орнату үшін керек тілдерді таңдаймыз (Сурет 3.24).



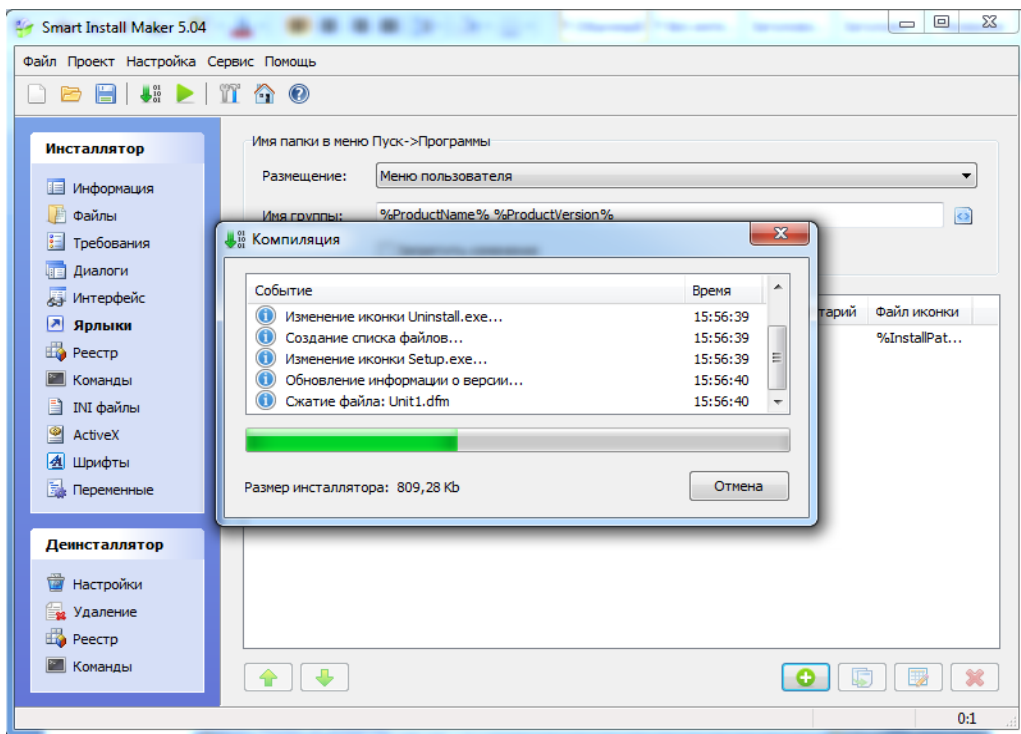
Сурет 3.24 – Тілдерді таңдау

«Ярлыки» батырмасын басып, керекті мәліметтерді қосамыз (Сурет 3.25).

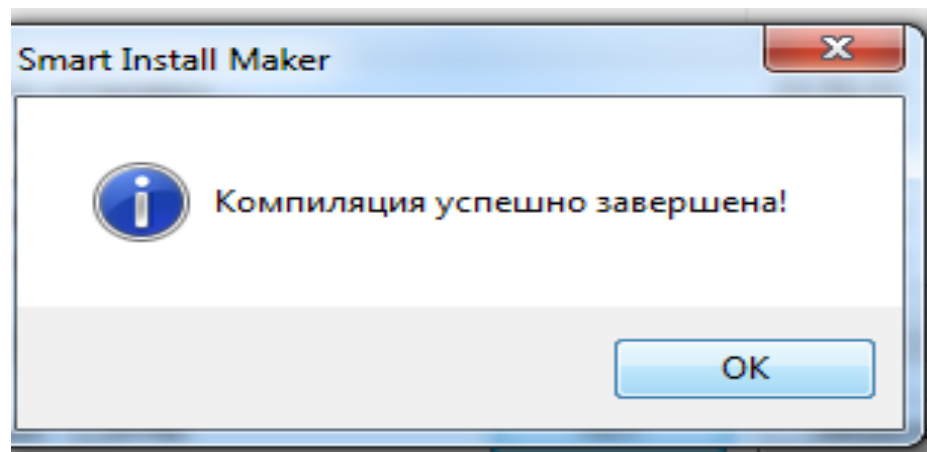


Сурет 3.25 – Ярлык құру

Одан кейін «компиляция» батырмасын басып, орнату дискін құруға жібереміз (3.26 - Сурет).



Сурет 3.26 – Компиляция жасау



Сурет 3.27 – Компиляция аяқталды

4 Тіршілік қауіпсіздігі

4.1 Жұмыс сипаттамасы

Менің дипломдық жобамда «Ағылшын тілін үйренуге арналған бағдарлама және Oxford 3000 сөздігі» бағдарламасын жасау қарастырылған. Бұл жобаны жасау кезінде компьютер негізгі рөл атқарады. Бұл бағдарлама Алматы қаласында ағылшын тілін үйрету орталықтарына және компьютерлік сыныптарға арналған. Менің талдауы бойынша бөлмеде сегіз компьютер, аптасына бес кун сегіз сағат жұмыс жұмыс істейді. Қолданушылар компьютермен жұмыс істеген кездегі еңбектің қауіпсіздігін сақтауымыз керек. Мекеме бөлмесінің сызбасы 4.2- суретте көрсетілген. Бөлменің ұзындығы 8 м, ені 6 м және биіктігі 3 м және үш терезе.

4.2 Жұмыс орнының микроклимат жағдайы

Бұл бөлмедегі метеорологиялық жағдайлардың кешені: температура, салыстырмалы ылғалдылық, аэроиондардың саны, ауа ауысымы, ауа қозғалысының жылдамдығы, ауада қатты бөлшектердің (шаң) болуы жағымды иістердің болуы (ароматерапия) және т.б. Тұрғын үй және қоғамдық мекемелерде жылдың жылы мезгілінде микроклимат үшін оңтайлы болып табылады: ауа температурасы 22-25 С, салыстырмалы ылғалдылық 30-60%, ауаның қозғалу жылдамдығы 0,25 м/с тан жоғары емес; жылдың суық мезгілінде бұл көрсеткіштер сәйкес 20-22° С, 30-45 % және 0,1-0,15 м/с құрайды. Мұнда терезелерден қарама-қарсы қабырғаларға горизонтал бойынша айырмашылығы 2С тан аспауы тиіс, ал вертикал бойынша мекеменің әрбір биіктік метріне 1С.

Мекемеді ыңғайлы микроклиматты құру үшін арнайы жүйелер қолданылады: суық мезгілдер – жылыту жүйесі: газдық, отын немесе көмір пештері; орталықтандырылған су жүйесі немесе сирек жағдайларда булық жылу; калориферлер мен электржылытқыштар. Жылдың жылы және ыстық мезгілінде – желдету және салқындату жүйелері.

4.3 Бөлменің өрт қауіпсіздігі және электр қауіпсіздігі

Бізде электр қауіпсіздігін қарастыру үшін МЕСТ 12.1.030-81 бойынша жерлендіру жасау жұмысын қарастыру керек. Бұл жерде кернеу - 220В болады, сол себепті жерлендіру немесе нөлдеу қажет етпейді, бірақ ұсынылады.

Құрылыс үшін темір бетон, әйнек, кірпіш, метал қолданылады. Оған қоса жанбайтын заттардан жасалған өртке қарсы өткелдерді ескеру керек, олар бөлмелердің арасында қойылады. Ал өрт крандарын болса дәлістерде немесе кіре беріс аумақтарына орнатамыз.

4.5 Шудың әсерін зеріктеу

Көрермендер залы және аппараттық бөлме арасында әуе шуының изоляция индексі 50дБ-ден кем болмағаны дұрыс. Аппараттық бөлме қабырғалары және төбесі 500 - 2000Гц жиілікте дыбысты жұту коэффициенті 0,6-дан кем болмайтын дыбыс жұтатын заттармен қапталуы қажет. Дыбысты қамтамасыз ету жүйесінің аппараттық бөлменің едендері күнделікті ылғалды жинастыру жұмыстарын жүргізуге мүмкіндік беретін материалдардан жасалуы.

Дербес компьютер қолданушысының, бөлмедегі шу көздері – сөйлесіп тұрған адамдардың, дербес компьютердің, кондиционердің шуы болып табылады.

Дыбысты жақсы жұтатын материалдар: фибролитті плиталар, минералды мақта, жарықталшықтарында, поливинилхлоридте болады.

4.6 Бөлме жоспары

Жұмыс орнының мінездемесі:

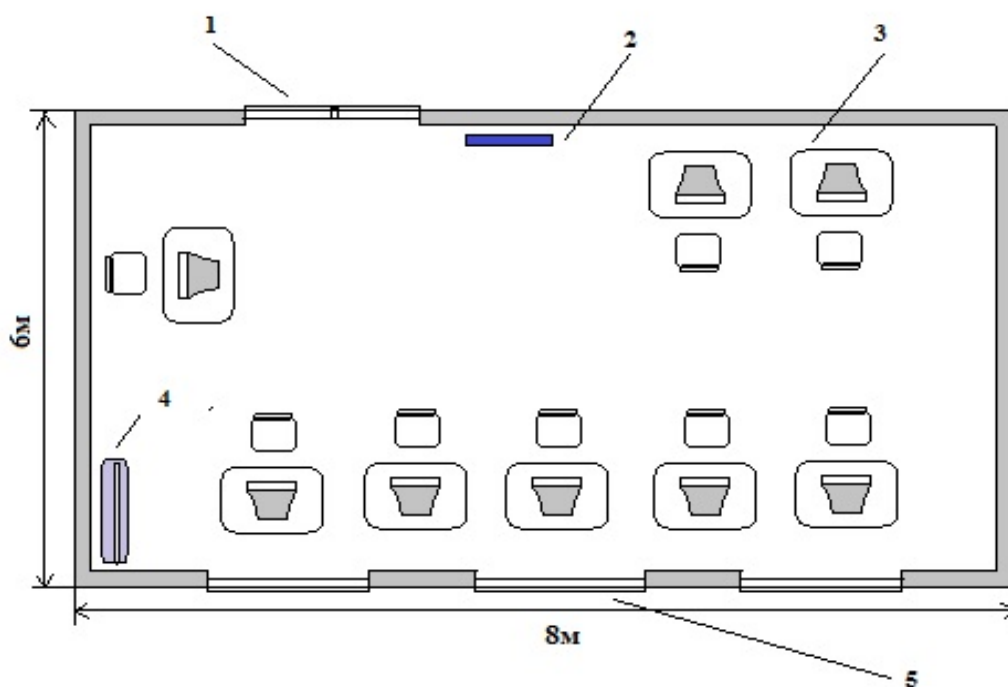
Бұл жұмыс орны ғимарат ішінде орналасқан, сыртқы шу байқалмайды.

- бөлменің ұзындығы 8 м, ені 6 м, биіктігі 3 м;
- жарық өткізгіш материал – екіқабатты шыны парағы;
- күн сәулесінен қорғайтын жабдықтары – жиналатын жалюздер;
- қабырға – ақ түсті;
- есік 1*2;

Компьютердің техникалық сипаттамалары:

- LG core i7 /8Gb/750Gb/DOS компьютері;
- LG LS17AN монитору;
- мөлшерлер 1100x650x1100 мм (үстел);
- электрлік көзі: қуаты 400 Вт, 220-250 В, 50 Гц жиілігі.

2 Сплит-жүйе Panasonic кондиционер, қуаты 5 кВт.



1 – есік, 2 – өрт сөндіргіш, 3- компьютер, 4-кондиционер, 5 – терезе

Сурет 4.2 – Бөлменің жоспары

4.7 Кондиционерлеу жүйелерін есептеу

Ауаны кондиционерлеудің талаптары әр түрлі саладағы технологиялық үрдістерді жүргізуде, компьютердің, басқа құралдардың жұмыс істеу мүмкіндігін қамтамасыз ету үшін ауа ортасының параметрлерін өндіріске сай қолдау қажет. Тағайындалуы бойынша ауа алмасу жүйесі қолайлы және технологиялық деп бөлінеді.

Бізге қажетті кондиционерді таңдауға ең алдымен артық жылудың қосындысын және оған күн сәулесінен бөлінетін жылу, жұмыс істейтін адамдардың санын, қолданылатын техникаларды есептеу керек. Бөлмедегі жылулықтың балансын табу төмендегі формула бойынша есептеледі

$$Q_{\text{жылу.б}} = Q_{\text{қоршау}} + Q_{\text{р}} + Q_{\text{а}}^{\text{а}} + Q_{\text{жарықтану}} + Q_{\text{құрал}}, \text{ Вт} \quad (4.1)$$

Бұл жерде $Q_{\text{қоршау}}$ – температуралар айырымынан алынатын жылу жоғалту;

$Q_{\text{р}}$ – шыны арқылы күннің радиациясы себебінен келетін жылулар;

$Q_{\text{а}}^{\text{а}}$ – адамдардан келетін жылулар;

$Q_{\text{жарықтану}}$ – жасанды жарықтандырудан келетін жылулар;

$Q_{\text{құрал}}$ – басқа техникалық аспаптардан келетін жылу.

Терезе арқылы күн сәулесінен келетін жылу.

Күннен сәулесінен келетін жылу әйнек түріне байланысты 90%-ға дейін жұтылады, қалғаны шағылып кетеді. Ең үлкен жылу шашырай түсетін және тура күн радиациясының ең жоғары деңгейінде болады. Күн сәулесінің түсуі жыл мезгіліне және де тәулік уақытына байланысты болады.

Салқын мезгілде сыртқы температура ($t_{\text{сырт.есеп}}$) ең салқын айдың 13 сағатындағы орта температурасына, жылы болса – ең ыстық айдың 13 сағатындағы орта температураға сәйкес болады.

$$Q_{\text{коршау}} = V_{\text{бөлме}} X_0 (t_{\text{шыққан}} - t_{\text{келген}}), \text{ Вт} \quad (4.2)$$

Мұнда $V_{\text{бөлме}}$ – бөлме көлемі, м^3 $V_{\text{бөлме}} = 8 \times 6 \times 3 = 143 \text{ м}^3$;

X_0 – меншікті жылулық сипаттама, $\text{Вт}/\text{м}^3\text{°C}$;

$X_0 = 0,42 \text{ Вт}/\text{м}^3\text{°C}$;

$t_{\text{сырт.есеп}} = 26,6^\circ$ - жылы мезгілге арналған сыртқы температурасы;

$t_{\text{сырт.есеп}} = -24^\circ$ - суық мезгілге арналған сыртқы температурасы;

$t_{\text{іш.есеп}} = 24^\circ$ - жылы мезгілге арналған ішкі температурасы;

$t_{\text{іш.есеп}} = 21^\circ$ - суық мезгілге арналған ішкі температурасы.

Жылы мезгілде

$$Q_{\text{коршау}} = 125 \times (27,6 - 23) \times 0,32 = 190,512 \text{ Вт}$$

Салқын мезгілде

$$Q_{\text{коршау}} = 125 \times 0,42 \times (-24 - 23) = -2381,4 \text{ Вт}$$

Шынылау нәтижесінде күн сәулеленуінен болатын жылу.

Күннің сәулеленуінен болатын жылу әйнек арқылы сәуле бөлмеге кіріп, шынылау сәулелену периоды

$$Q_p = (q_{\text{тура}} + q_{\text{шашыр.}}) K_1^c K_2 \beta_{\text{ж.ө.}} n H_0 V_0, \text{ Вт} \quad (4.3)$$

Күн сәулесі көлеңке периоды үшін (шашыраңқы радиация)

$$Q_p = q_{\text{шашыр.}} K_1^T K_2 \beta_{\text{ж.ө.}} n H_0 V_0, \text{ Вт} \quad (4.4)$$

Бұл жерде $q_{\text{тура}}$; $q_{\text{шашыр.}}$ – тура және шашыраңқы радиациядан келетін жылулық ағындар, $\text{Вт}/\text{м}^2$;

$F_0 = n H_0 V_0$ – жарықтық ойы қауданы, м^2

K_1 – шынылаудың көлеңкеленуі (K_1^C – сәулеленген ойықтар; K_1^T – көлеңкедегі ойықтар);

K_2 – ластану коэффициенті;

$\beta_{ж.ө.}$ – жылу өткізгіштік коэффициенті.

1 Менің жобамда шынылаудың ауданы, 43^0 СШ [12, кесте 3] $F_0 = 2 \times 2,5 \times 2 = 12 \text{ м}^2$.

2 Шынылау бағыты: оңтүстік-шығыс.

3 Ішінде жарық перделері бар. $\beta_{ж.ө.} = 0,3$ [12, кесте 4] деп аламыз.

Түске дейін оңтүстік-шығыс үшін, яғни сағат 9-дан 13-ге дейін 43^0 СШ ендікте тура сәуле мәні (П) $q_{тура} = 334 \text{ Вт/м}^2$ және шашыраңқы сәуле мәні (Р) $q_{шашыр.} = 101 \text{ Вт/м}^2$ тең [12, кесте 5]. $43-66^0$ СШ ендік диапозонда металды екі қабатты шынылау жүргізу үшін: $K_1 = K_1^C = 0,72$, 9-10 және 12-13 сағат аралығындағы период үшін. $K_1 = K_1^T = 1,15$, 13-16 және 18-21 сағат аралығындағы период үшін. Әйнектің ластану коэффициенті $K_2 = 0,9$.

Тура сәулеленуде 9 бен 14 сағат аралығында есептеу төмендегі формуламен жүреді (4.12)

$$Q_p = (388 + 101) \times 0,72 \times 0,9 \times 10 \times 0,4 = 1265 \text{ Вт}$$

Көлеңкелеу периодында 14 пен 20 сағат аралығында осы формуламен есептейміз (4.13)

$$Q_p = 22 \times 1,15 \times 0,9 \times 10 \times 0,4 = 91 \text{ Вт}$$

Жоғары есептелу уақыты: 9-10 сағат, жылу түсу 1265 Вт.

Адамдардан шығатын жылу.

Адамдардан шығатын жылу ауа параметрлеріне және орындалатын жұмыс көлеміне қарай. Адамдардың жылу таратуы 4.3– кестемен сипатталады

К е с т е 4.3 –Адамның сыртқы ортаға жылу таратуы, Вт

Сыртқы орта температурасы °С	Отырғандағы жағдай			Тұрғанда немесе жеңіл қозғалыс			Ауыр жұмыс		
	Анық	Жасырын	Жалпы	Анық	Жасырын	Жалпы	Анық	Жасырын	Жалпы
24	67	35	102	72	60	132	95	154	249
20	82	21	103	92	42	133	140	110	250

К е с т е 4.4 – Адам бөлетін ылғал және көміртегі саны

Параметрлер	Бөлме ауасының температурасындағы мәндер °С				
	15	20	25	30	35
Ылғал г/сағ	40	40	50	75	115
Көміртегі қостотығы г/сағ	45	45	45	45	45

Бөлмеде 8 қолданушы отырады. $t = 24$ °С температурада отырған күйде ер адам 67 Вт жылу бөледі [12, кесте 8]. Әйел адам болса ер адамның жылу бөлу нормасының 85 %-ын, ал кішкентай бала болса – 75 %-ын бөледі. Бөлмеде адамдардың бөлетін жылуы: $Q_a^a = 67 \times 4 \times 0,85 = 227,8$ Вт. Ал жалпы жылуы болса $Q_a^ж = 102 \times 4 \times 0,85 = 346,8$ Вт.

$t = 24$ °С есептеу үшін ылғалдылық және көміртегі қышқылы мәндерін 9-кестеден [12] аламыз: бір адамнан 50 г/сағ ылғалдылық, 45 г/сағ көміртегі қышқылы бөлінеді екен. Ал 8 адамның ылғалдылығы $8 \times 50 = 400$ г/сағ, көміртегі қышқылы мөлшері $8 \times 45 = 360$ г/сағ құрайды.

$t = 20$ °С үшін: 1 адамнан шығатын ылғалдылық – 40 г/сағ, көміртегі қышқылы – 45 г/сағ. 8 адамнан бөлінетін ылғалдылық: $8 \times 40 = 320$ г/сағ. 8 адамнан шығатын жалпы көміртегі қышқылы: $8 \times 45 = 360$ г/сағ.

К е с т е 4.5 – Адамдардан бөлінетін зиянды заттардың есептелуінің нәтижелері

Жыл мезгілі	Температура °С	Жылу, Вт		Ылғалдылық, W г/сағ	CO ₂ г/сағ
		Q _a ^a	Q _a ^ж		
Жылы	24	227,8	346,8	200	180
Салқын	20	278,8	350,2	160	180

Ауа алмасуына есептеу жүргізу

$Q_{жылу.б}$ жазда $>$ $Q_{жылу.б}$ қыста, $Q_{жылу.б}$ жазда ауаның жылу кернеулігін төмендегі формуламен есептейміз

$$Q_k = \frac{Q_{жылу.б} \cdot 860}{V_{бөлме}} = \frac{4,573 \cdot 860}{7 \cdot 6 \cdot 3} = \frac{31,21 \text{ ккал}}{\text{м}^3} \quad (4.6)$$

$$Q_n > 20 \text{ ккал/м}^3 \text{ болғанда } \Delta t = 8^\circ \text{C}$$

Бөлмеге керекті ауа мөлшері жылулық балансынан алынып, төмендегі формула бойынша анықталады

$$L = \frac{Q_{\text{жылу}} \cdot 860}{C \cdot \Delta t \cdot \gamma} = \frac{4,573 \cdot 860}{0,24 \cdot 8 \cdot 1,206} = 1698 \text{ м}^3 / \text{сағ} \quad (4.7)$$

мұнда $C = 0,24$ ккал/кг⁰С - ауаның жылу сыйымдылығы;
 $\gamma = 1,206$ кг/м³ - ағынды ауаның массасы.

Артық жылулар 16462,8 кДж/сағ. Бөлмеге керекті ауа $L = 1698 \text{ м}^3 / \text{сағ} = 28,3 \text{ м}^3 / \text{мин}$. Біз есептеулерге сүйене отырып өз таңдауымызды Panasonic CS/CU-E 18 кондиционеріне тоқтаттық.

Кондиционердің жалпы техникалық сипаттамасы:

- EER, A 4,36 ;
- COP, A 4,41;
- жылдық электр ток пайдалануы 945 кВт*сағ
- шу деңгейі ішкі (жоғ/орт/төм), дБ(А) 45/36/33, сыртқы, дБ(А) 48;
- салқын 5,30кВт; жылу 5,90 кВт;
- электр қорек кернеуі 220В,50 Гц;
- салқынның жұмсайтын қуаты, кВт 1,45;
- жылудың жұмсайтын қуаты, кВт 1,52
- салмағы 13 кг.

4.8 Бөлменің вентиляция жүйесін есептеу

Өндірісте динамикалық климатты бір деңгей ауысуымен бірге жасауды ұсынады. Вентиляция жүйесінің характеристикасын сипаттайтын басты параметрлері, еселік айырбастауы болады, бұл ауаның бөлмедегі қаншы рет ауысқанын білдіреді.

Ауа еселегін бір терезені есептейік, терезенің өлшемдері 2 м x 1 м болып келеді. Бөлмеде сегіз адам жұмыс істейді. 5.6-кестедегі электр жабдығықтарды қолданамыз.

5.6-к е с т е – Қолданылатын электржабдықтары

Атауы	Саны (дана)	Қолданылатын қуат, Вт
Дербес компьютер	8	600
Тоқтаусыз қоректендірі көзі	8	700

Ауа ауысуы еселегін мына формуламен табамыз

$$K = \pm \frac{V_{\text{вент}}}{V_{\text{бл}}} \quad (4.11)$$

мұнда, $V_{\text{вент}}$ – бөлмеге келетін ауа көлемі, м³;

$V_{\text{бл}}$ – бөлме көлемі, м³.

Ауысуға қажетті ауа көлемін $V_{\text{вент}}$ жылулық тепе-теңдік бойынша есептейміз.

$$V_{\text{вент}} = \frac{3600 \cdot Q_{\text{аж}}}{C \cdot (t_{\text{кт}} - t_{\text{кл}}) \cdot Y} \quad (4.12)$$

Бұл жерде $Q_{\text{аж}}$ – артық жылу, Вт;

$C = 1000$ – ауа жылуөткізгіштілігі, Дж/кг·К;

$Y = 1.2$ – ауа тығыздығы, кг/м³.

Тыс кететін ауа температурасы осы формула бойынша анықталады

$$t_{\text{кт}} = t_{\text{км}} + (H - 2) \cdot t \quad (4.13)$$

мұнда t – бөлменің 1 метр биікке температура өзгеруі, 1 градустан ден 5ке дейін мәндерін аламыз.

$t_{\text{км}}$ – жұмыс орында температура, 23 градус.

H – бөлме биіктігі, 3 метр.

$t_{\text{кл}}$ – келетін ауа температурасы, 18 градус.

5.12-формула бойынша ауа температурасы мынаған тең

$$T_{\text{кт}} = 23 + (3 - 2) \cdot 2 = 25.$$

Артық жылуын төмендегі формула бойынша анықтаймыз

$$Q_{\text{аж}} = Q_{\text{аж1}} + Q_{\text{аж2}} + Q_{\text{аж3}} \quad (4.14)$$

мұнда $Q_{\text{аж1}}$ – электоржабдықтардан артық жылу, Вт;

$Q_{\text{аж2}}$ – күннің радиациясынан келген жылу, Вт;

$Q_{\text{аж3}}$ – адмамның жылу беруі, Вт.

Электржабдықтардан артық жылуын мына формулаарқылы есептейміз

$$Q_{\text{аж1}} = E \cdot P \quad (4.15)$$

мұнда E – жылу бұру энергиясының шығын коэффициенті;

P – электр жабдықтар қуаты, Вт.

Элетрэнергия шығыны келесі мәндерді қабылдайды

$E=0.4$ дербес компьютер үшін;

$E=0.55$ жарықтандыру үшін.

Төмендегі формула бойынша электр жабдықтардың және жарықтандырудың жалпы қуатын есептеп аламыз

$$P = P_{\text{дана}} K \quad (4.16)$$

Бұл жерде $P_{\text{дана}}$ – бір жабдықтың қуат өлшемі, Вт;

K – жабдықтардың сан данасы.

Содан , 5.16- формула бойынша

$$P_{\text{жарықтандыру}} = 200 \cdot 8 = 1600 \text{ Вт}$$

$$P_{\text{компьютеры және т.б.}} = 600 \cdot 8 + 700 \cdot 8 = 10400 \text{ Вт}$$

5.15- формула бойынша жарықтандырудан және электржабдықтардан артық жылуын есептейміз

$$Q_{\text{аж1}} = 0.55 \cdot 1600 + 0.4 \cdot 10400 = 5040 \text{ Вт.}$$

Жасанды жарықтандырудан артық жылу келесі формула бойынша есептейміз

$$Q_{\text{аж3}} = m \cdot S \cdot k \cdot Q_{\text{T}} \quad (4.17)$$

мұнда m – терезе саны;

S –терезе ауданы, м^2 ;

k - әйнектеу коэффициенті. Екі әйнектеу коэффициенті – $k = 0.8$

Q_{T} – терезеден жылу келіуі, 127 Вт/м тең.

5.17-формула бойынша жасанды жарықтандырудан артық жылуын есептеймік

$$Q_{\text{аж2}} = 3 \cdot 2 \cdot 0.8 \cdot 127 = 610 \text{ Вт.}$$

Адамның жылу беруің келесі формула бойынша есептейміз

$$Q_{\text{аж3}} = n \cdot q \quad (4.18)$$

мұнда q –бір адамның жылуы бөлгіштігі, 80 Вт/ад тең;

n – адам саны, 8 тең.

5.18-формула бойынша адамның жылу беруін есептейміз

$$Q_{аж3} = 8 \cdot 80 = 640 \text{ Вт.}$$

5.12-формула бойынша артық жылуды есептеп аламыз

$$Q_{аж} = 5040 + 610 + 640 = 6290 \text{ Вт.}$$

Сөйтіп, 5.12-формула бойынша ауысуға керекті ауа көлемін есптейміз

$$V_{вент} = \frac{3600 \cdot 6290}{1000 \cdot (25 - 18) \cdot 1,2} = 2695,71$$

Демек, оптималды шартты тудыру үшін ауаның вентиляция жүйесін жүзеге асыру керек.

4.9 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды

Тіршілік қауіпсіздігі бөлімінде бөлмедегі жұмыс шарты және микроклимат параметрлерін талдау және ауаны кондиционерлеу, бөлменің вентиляция жүйесіне анализ жасалды. Еңбек шарты деңгейі өте дұрыс, есептеу нәтижесінде алынған жауаптар, тіршілік қауіпсіздік бойынша стандарттарды қанағаттандырады.

Кондиционерді таңдау үшін ең бірінші артық жылу қосындысын, және оған күннің сәулесінен бөлінетін жылу, жарықтандыруды, жұмыс істейтін адамдар санын, басқада бөлмеде қолданылатын техникаларды есептеу қажет.

Есептеп, қорыта келгенде бөлмеге керекті ауа мөлшері $L = 1698 \text{ м}^3/\text{сағ} = 28,3 \text{ м}^3/\text{мин}$. Осы талаптарды толық қанағаттандыратын Panasonic CS/CU-E 18 кондиционерін таңдадым.

5 Техника-экономикалық негізделуі

5.1 Жұмыстың сипаттамасы

«Ағылшын тілін үйренуге арналған бағдарлама және Oxford 3000 сөздігі» бұл компьютерге арналған ағылшын тілін дамытуға арналған қосымша бағдарлама. Бұл арнайы бағдарлама Embarcadero RAD Studio Delphi 2010 бағдарламасымен Microsoft SQL деректор қорын байланыстыру арқылы жасалады.

Бұл бағдарламаның негізгі артықшылықтары ағылшын тілін үйренуге арналған сабақтар, білімін тексеруге арналған тесттер, кестелер, және қосымша ағылшан тілінде ең көп қолданылатын сөздер «Oxford 3000 сөздігі» бар.

5.2 Маркетингтік стратегия

Маркетинг-микс – Маркетингтік құралдар жиынтығы, бұл арқылы кез-келген компания өзінің тауарлары немесе қызметтеріне сұраныс арттыра алады. Белгілі бір тауарға белгілі баға бойынша белгілі бір орында жұмыс жасау арқылы сұраныс қалыптастыратын стратегия жүйесі.

Маркетинг-миксті төрт тактикалық әрекет бойынша сипаттауға болады: Маркетинг-микс жиындарының міндеті – бір нарықта потенциалды пайдаланушылардың қажеттерін өтеумен ғана шектелмейді, оған қоса ұйымның әсерін көтеруге тырысатын жиынтықтарды құрастыру.

Қозғалу (Promotion) – компанияның тауарлары туралы ақпарат таратуы осы тауарларды сатып алуға шақыратын мүмкіндігі зор жарнама, іс-шаралар.

Продукт (Product) – компанияны біртұтас нарыққа ұсынатын «тауарлар» жиыны.

Дистрибуция (Place) – тауар бірыңғай пайдаланушыларға қолжетімді болу үшін іске асырылатын көп іс-шаралар.

Баға (Price) – пайдаланушылардың тауарды сатып алу үшін төлеуі керек ақша.

Кесте 5.1. Маркетинг-микс элементтері

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Product	
Ағылшын тілін дамытуға арналған бағдарламаның қызметі	Жалпы тапсырманы шешу келесі бөлімдерден тұрады: 1) Пайдаланушы бағдарламада арнайы сабақтар кестелер арқылы ағылшын тілін қайталай алады, арнайы тест арқылы білімін тексеріп көруіне болады және қосымша

Кесте 5.1 соңы

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Promotion	

	Басында уақытша жарнамасыз жұмыс істеу, яғни тікелей сату
Place	
№ 1 оқу орталығы, № 2 оқу орталығы, № 3 оқу орталығы, № 4 оқу орталығы, № 5 оқу орталығы,	Алматы қаласындағы ағылшын тілін үйрететін орталықтардың администраторларын хабарландыру, келіссөз жүргізу

5.2.1 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі

Еңбекке кеткен шығынның көрсеткіштері осы формула бойынша есептеледі

$$Q=q*c \quad (5.1)$$

Мұндағы q - бағдарламалық өнімдегі операторлар саны, менің жұмысымда операторлар саны 3000-ға тең. c -бағдарлама күрделілігінің коэффициенті, ол менде 1.26 ге тең. Базалық көрсеткішті табамыз [1]

$$Q=3000*1.26=3780$$

Бағдарламалық өнімді жасауға кететін уақытты есептеу керек. Уақыт адам-сағатпен есептеледі, ал T_d нақты істелген уақытпен алынады, ал қалған кезеңнің уақыты Q командасының шартты санына байланысты есептік жолмен анықталады. Бағдарламаны жасауға кеткен әр кезеңнің уақытын табамыз

$T_{по}$ (мақсат сипатын дайындау уақыты), нақты деректер бойынша алынады және мынаған тең деп алынады (3-тен 5 күнге дейін, 8 сағаттан)

$$T_{по} = 24 \text{ адам} / \text{сағ}$$

T_o (мақсатсипаттамасы уақыты) келесі формуламен анықталады

$$T_o = Q \times B / (50 \times K) \quad (5.2)$$

$$T_o=3780*1.2/50*0,8=113,4 \text{ адам/сағ}$$

мұндағы B – мақсат есебі өзгерісінің коэффициенті, K коэффициенті мақсат күрделілігіне және өзгеріс санына тәуелді – 1,2-ден 1,5-ке дейін

K – бағдарлама дайындаушы біліктілігінің коэффициенті, менің берілген бұл салада жұмыс істеу өтілім 2 жылға дейінгі уақыт болғандықтан, K коэффициент мәні – 0,8-ге тең болады.

T_A (алгоритм құруға кеткен уақыт) мына формуламен есептейміз

$$T_A = Q / (50 \times K) \quad (5.3)$$

$$T_A = 3780 / (50 \times 0,8) = 94,5 \text{ адам/сағ}$$

T_{BC} (блок – сұлба құруға кеткен уақыт) T_A сияқты 3 формуламен есептеледі.

T_H (бағдарламаның тілінде жазуға кеткен уақыт) келесі формуламен анықталады

$$T_H = Q \times 1,5 / (50 \times K). \quad (5.4)$$

$$T_H = 3780 \times 1,5 / (50 \times 0,8) = 141,75 \text{ адам/сағ}$$

T_{II} (бағдарламаны теріп шығу уақыты мына формуламен анықталады:

$$T_{II} = Q / 50 \quad (5.5)$$

$$T_{II} = 3780 / 50 = 75,6 \text{ адам/сағ}$$

1) T_{OT} (бағдарламаны тестілеу уақыты) келесі формуламен анықталады:

$$T_{OT} = Q \times 4,2 / 50 \times K = 1600 \times 4,2 / 50 \times 0,8 = 396,9 \text{ адам/сағ} \quad (5.6)$$

8) T_D (құжаттарды рәсімдеу уақыты), нақты деректер бойынша алынады (3-тен 5 күнге дейін, күніне 8 сағат)

$$T_D = 24 \text{ адам / сағ}$$

Еңбек шығындарының сомасы еңбек шығынының құрама сомасы ретінде 7 формуламен есептеледі

$$T = T_{II} + T_{TO} + T_A + T_{BC} + T_H + T_{II} + T_{OT} + T_D. \quad (5.7)$$

$$T = 24 + 113,4 + 94,5 + 94,5 + 141,75 + 75,6 + 396,9 + 24 = 964,65 \text{ адам/сағ}$$

Бағдарламалық қамтаманы өңдеуге кеткен шығынның есептелуі

Еңбек құны екі құрамнан құралады: негізгі еңбек құны және қосымша еңбек құны. Негізгі еңбек құны

$$Z_{\text{осн}} = t_{\Sigma} * TC / (t_{\text{cp}} * 8), \text{ теңге} \quad (5.8)$$

мұндағы t_{Σ} – (7) формуласы бойынша есептелетін жалпы еңбек шығыны;

t_{cp} – бір айдағы орташа күн саны, 21 күнге тең, жұмыс күніндегі 8 сағатқа көбейтеміз.

Минималды еңбек ақы 01.01.2014ж бастап ҚР-да 19966 теңгені құрады.[2]

TC – тарифтік қойылым = $19966/8 = 2495,75$ ке тең болды.

t_{Σ} – (7) формуласы бойынша 964,65 –ге тең.

Осылайша негізгі еңбек ақы:

$$Z_{\text{осн}} = 964,65 * 2495,75 / 21 = 114644 \text{ теңге}$$

Қосымша еңбек ақы негізгі еңбек ақының 20%-ын құрайды. (5.9) формуласы бойынша анықталады

$$Z_{\text{доп}} = 0,2 * Z_{\text{осн}} = 22928,8 \text{ теңге}$$

Жалпы еңбек ақы (еңбекті төлеу фонды) негізгі және қосымша еңбек ақы қосындысымен анықталады

$$\text{ФОТ} = Z_{\text{доп}} + Z_{\text{осн}} = 22928,8 + 114644 = 137572,8 \text{ теңге} \quad (5.10)$$

5.2.2 Қосымша шығындар статьясы

Бағдарламалық қамтамасыз етудегі шығындар бағдарламаға кеткен шығындар, әлеуметтік сақтандыруға аударымдар, жүккұжатқа кеткен шығындар, амортизациялық аударымдар, техникалық қондырғылар қызмет көрсетуге кеткен шығындар, компьютермен жұмысына кеткен электроэнергиялық шығындар.

ЭЕМ - на бағдарлама жазу үшін құрылғы ретінде дербес компьютер қолданылады. Бағасы $C_{\text{обор}} = 189000$ теңге.[3]

Материалдар мен көмекші бөлшектер шығыны, бағдарламалық өнімді жазу барысында қолданылды ($C_{\text{мжк}}$), сонымен қатар техникалық қызмет көрсету шығыны ($C_{\text{то}}$), құрылғы құнынан 1.5% және 2.5% [4];

$$C_{\text{мжк}} = 0,015 * C_{\text{обор}} = 0,015 * 189000 = 2835 \text{ теңге} \quad (5.11)$$

Амортизациялық аударымдар өнімнің физикалық жарамсыздығына байланысты сол өнімді толық қалпына келтіру үшін жиналатын ақша қоры.

Амортизациялық аударым амортизация нормасы бойынша орнатылады, (2.14) формуласы бойынша есептеледі

$$A = \frac{C_{\text{обор}} \times H_A \times N}{100 \times 12 \times t} \quad (5.12)$$

Мұндағы $C_{\text{обор}}$ – дербес компьютер құны; H_A – амортизация нормасы, N – жұмыс орындалуына кеткен күннің саны;
 t – компьютерді қолдануға кеткен уақыт.

$$N = T/24 = 964,65 / 24 = 40,19 \text{ күн}$$

T еңбек шығынына тең, оның мәні – 964,65 адам/сағ тең болған.
 Жалпы жұмыс істеу уақыты мына формула бойынша есептеледі

$$t = T_A + T_{BC} + T_H + T_{II} + T_{OT} = 94,5 + 94,5 + 141,75 + 75,6 + 396,9 = 803,25 \text{ адам/ сағ}$$

Амортизация нормасы мына формула бойынша анықталады

$$H_A = \frac{C_{\text{обор}} - C_{\text{ликв}}}{T_{\text{норм}} \cdot C_{\text{обор}}} \cdot 100\%, \quad (5.13)$$

мұнда $C_{\text{ликв}}$ – таратылым құны, жабдық құнынан 5% құрайды

$$C_{\text{ликв}} = 0,05 \cdot C_{\text{обор}} = 0,05 \cdot 189000 = 9450 \text{ теңге} \quad (5.14)$$

$T_{\text{норм}}$ – нормативті қызмет көрсету уақыты (компьютер үшін ҚР да $T_{\text{норм}} = 4 \text{ жыл}$) Осыдан

$$H_A = \frac{C_{\text{обор}} - C_{\text{ликв}}}{T_{\text{норм}} \cdot C_{\text{обор}}} \cdot 100\% = \frac{189000 - 9450}{4 \cdot 189000} \cdot 100\% = 23,75\%,$$

$$A = \frac{C_{\text{обор}} \times H_A \times N}{100 \times 12 \times t} = 189000 \cdot 23,75 \cdot 40,19 / 100 \cdot 12 \cdot 803,25 = 18,71 \text{ теңге}$$

Бағдарлама жазу кезіндегі электроэнергияның шығынын қоса есептеу керек.
 Электроэнергияның құны мына формуламен есептеледі

$$C_{\text{ЭЭ}} = M \cdot k_3 \cdot T \cdot C_{\text{кВт-ч}} \quad (5.15)$$

мұндағы M – ЭВМнің қуаты (450 Вт); k_3 – жүктеу коэффициенті (0.8); $C_{кВт.ч}$ – 1 кВт-сағ электроэнергия құны, оның мәні Алматы қаласында – да 14,36 теңгеге тең; [5]

$F_{эф}$ – жұмыс уақытының эффективті фонды, мына формула бойынша есептеледі (2.17):

$$F_{эф} = D_{ном} * d * (1 - f / 100) = 258 * 8 * (1 - 2,6 / 100) = 2010 \quad (5.16)$$

Мұндағы $D_{ном} = 258$ – бір жыл ішіндегі жұмыс күнінің саны; $d = 8$ – жұмыс күнінің ұзақтығы [сағ]; $f = 2.6\%$ – компьютерді жөндеуге кеткен уақыт.

Берілген параметрлер мен коэффициенттер бойынша электроэнергия құны

$$C_{ээ} = 0,45 * 0,8 * 2010 * 14,36 = 10390,8 \text{ теңге}$$

Амортизациялық аударымдар мен электроэнергия шығыны – жылдық шығын мәні, оны жылдық эксплуатациондық шығынмен анықталатын уақыт коэффициентімен дұрыстау керек.

$$\mathcal{E}_3 = t_{\Sigma} * C_3 / F_{эф}, \text{ теңге} \quad (5.17)$$

Мұндағы $C_3 = C_{ээ} + C_{ТО} + A_{жыл}$ – ЭЕМ-ның жалпы жылдық эксплуатационды шығынның құны
Амортизациялық аударылымдар

$$A_{жыл} = C_{обор} \cdot \frac{H_A}{100\%} = 189000 * \frac{23.75\%}{100\%} = 44887,5 \text{ теңге}$$

Техникалық қызмет көрсету шығыны

$$C_{ТО} = 0,025 \times C_{обор} = 0,025 * 189000 = 4725$$

$$C_3 = C_{ээ} + C_{ТО} + A_{жыл} = 10390,8 + 4725 + 44887,5 = 60003,3 \text{ теңге}$$

Мұнда $F_{эф}$ – жұмыс жасау уақытының эффективті фонды, (16) формула бойынша есептеледі, – t_{Σ} – ЭВМ-нің (7) формуласы бойынша анықталған есепті шешу үшін жалпы қолданылған уақыты, бұл жерде компьютердегі жұмыс уақытын ескеріп анықталған

$$t_{\Sigma} = t_{пр} + t_{отл} + t_d = 94,5 + 396,9 + 24 = 515,4 \text{ сағ}$$

ЭЕМ-ның жалпы жылдық эксплуатационды шығыны

$$\mathcal{E}_3 = 515,4 * 60003,3 / 2010 = 15385,9 \text{ теңге}$$

уақыт коэффициенті осы формула бойынша есептеледі

$$w = \frac{\Xi_3}{C_9} = \frac{15385,9}{60003,3} = 0,25 \quad (5.18)$$

Осылайша, жалпы эксплуатационды шығынды уақыт коэффициентін ескеріп дұрыстаймыз

– Электроэнергияға шығын

$$C_{\text{ЭЭ}}^* = C_{\text{ЭЭ}} \cdot w = 10390,8 \cdot 0,25 = 2597,7 \text{ теңге}$$

– Амортизационды аударым

$$A_{\text{жыл}}^* = A_{\text{жыл}} \cdot w = 44887,5 \cdot 0,25 = 11221,8 \text{ теңге}$$

(2.11) формула бойынша есептелетін жұмыс жалақысына тәуелді шығындар бар. Әлеуметтік салыққа кететін аударымдар жалпы жалақының 11% құрайды [11%, ҚР ның Салық Кодексі], (2.20) формуласы бойынша анықталады

$$CC = (\text{ФОТ} - 0,1 \cdot \text{ФОТ}) \cdot 0,11 = (137572,8 - 0,1 \cdot 137572,8) \cdot 0,11 = 13620 \text{ теңге} \quad (5.19)$$

Жүктеме шығындар қызмет көрсету мен басқаруға қатысты, құрылғының эксплуатациясы мен басқа да өндіріс процесін қамтамасыз ететін қосымша шығындар болып табылады, еңбек жалақысы фондының 50% құрайды, (20) формуласы бойынша анықталады

$$C_{\text{накл}} = 0,5 \cdot \text{ФОТ} = 0,5 \cdot 137572,8 = 68786,4 \text{ теңге} \quad (5.20)$$

5.2.3 Өзіндік құн нәтижесінің кестесі

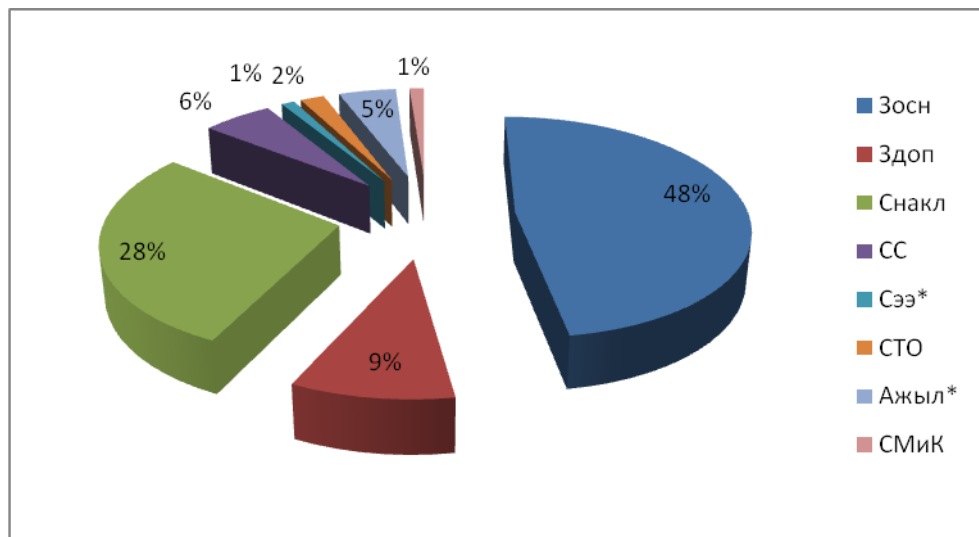
Бағдарламаны өңдеуге кеткен бүкіл шығын еңбек жалақысы фондының, эксплуатационды шығынның, әлеуметтік сақтандыру, жүктеме шығындар мен материал және құрамалардың қосындысына тең болады (Сурет 5.1).

Бағдарламаны өңдеудің қорытынды құны 5.2 кестеде.

Кесте 5.2 – Өзіндік құн кестесі

Шығын статьялары		Сумма, теңге	Жалпы суммадан Процентпен
ФЗП	$Z_{\text{осн}}$	114644	47,49

	$Z_{доп}$	22928,8	9,49
Жүктеме шығындар, $C_{накл}$		68786,4	28,49
Әлеуметтік сақтандыру, CC		13620	5,64
Эксплуатационды Шығындар	$C_{ээ}^*$	2597,7	1,07
	$C_{ГО}$	4725	1,95
	$A_{жыл}^*$	11221,8	4,64
Материалдар мен комплектілер, $C_{мик}$		2835	1,17
Барлығы:		241358,7	100



Сурет 5.1 – Өнімнің өзіндік құнының диаграммасы

5.2.4 Бағдарламаның бағасына есептеу

Өзіндік құны 241358,7 теңге болды, онда бір енгізу объектісіне арналған жүйенің толық өзіндік құны (C_{nc})

5 компанияға сататын болғандықтан 5 ке бөлеміз

$$C_{nc} = 241358,7 / 5 = 48271,74 \text{ теңге}$$

Өнімнің минималды бағасы келесі формуламен есептеледі

$$C_{min} = (C_{nc} + C_{мд} + C_{з}) (1 + r/100) = (48271,74 + 180 + 150) * 1,21 = 58808,1 \text{ теңге} \quad (5.21)$$

Мұндағы $\Pi_{\text{ндс}}$ - диск бағасы, қазіргі нарықтық бағасын қарастырып, 180 теңге деп алынды;

Π_3 – жазба бағасы;

r - Өзіндік құнға қатынасымен алынған тиімділіктің жоспарланған пайызы, ол $r = 21\%$;

НДС ті есептегенде минималды баға осы формуламен есептеледі (ҚР да НДС 12%)

$$\Pi_{\text{minНДС}} = \Pi_{\text{min}} + \Pi_{\text{ндс}} \quad \Pi_{\text{min}} = 58808,1 + 0,12 * 58808,1 = 65865,07 \text{ теңге} \quad (5.22)$$

Кіріс төмендегі формуламен бойынша есептеледі

$$V_{\text{реал}} = \Pi_{\text{minНДС}} * V_{\text{объем}} = 65865,07 * 5 = 329325,36 \text{ теңге} \quad (5.23)$$

Мұнда $V_{\text{объем}}$ – бағдарламанық құралды жүзеге асыру көлемі;

Жалпы пайда төмендегі формула бойынша есептеледі

$$P = V_{\text{реал}} - V_{\text{реал}} * \Pi_{\text{ндс}} - C_{\text{разраб}} = 329325,36 - 329325,36 * 0,12 - 241358,7 = 48447,26 \text{ теңге} \quad (5.24)$$

мұндағы НДС – қосылған құн салығы (12%)

Таза пайда келесі формуламен есептеледі

$$P_{\text{чист}} = P - N_{\text{прибыль}} = 48447,26 - 0,2 * 48447,26 = 38757,8 \text{ теңге} \quad (5.25)$$

Мұнда $N_{\text{прибыль}}$ – пайдаға салынатын салық (ҚР да $N_{\text{прибыль}} = 20\%$)

5.2.5 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу

Кәсіпорынның бірмезгілдік шығындары келесі бөлімдерден тұрады:

- жүйенің құны;
- көліктің шығыны;
- оқуға кететін төлем.

Жүйе құны 65865,07 теңге (22 формуласы).

Көлік шығындары жүйе құнының 20 пайызын құрайды

$$65865,07 * 0,2 = 13173 \text{ теңге}$$

Жұмысшыларды бағдарламамен жұмыс істеуге оқыту шамамен екі сағатта фирманың маманы көмегі арқылы сағатына 2500-3000 теңгеде болып табылады[3].

Сонда оқу бағасы:

$$3000*2=6000 \text{ теңге}$$

Кесте 2 – Компанияның бірмезгілдегі шығындарының есептеулері көрсетілген.

Шығындар түрлері	Сумма, теңге
Жүйе құны	65865,07
Көлік шығындары	13173
Жұмысшыларды оқыту бағасы	6000
Қорытынды:	85038,08

Негізгі шығын түрлері:

- жұмысшы еңбегіне төленетін ақшасы;
- әлеуметтік салықтар;
- басқада әр түрлі шығындар.

Дербес компьютер операторының жалақысы 50000 теңге

Бір жылдық жалақысын есептейтін болсақ 600000 теңге

Әлеуметтік салыққа жұмсалатын аударымдар(ҚР да әлеуметтік салық мөлшері 11% жұмысшы еңбегіне төленетін ақша қорынан, ҚР Салық Кодексі)

$$(600000-600000*0,1)*0,11=59400 \text{ теңге}$$

Электрэнергияға кететін шығындар

$$0,8*14,36= 11,48 \text{ теңге}$$

мұндағы 0,8 – жұмсалған қуат мөлшері, кВт-сағ;

14,36 – Алматыда 1 кВт-с электрэнергияның орташа құны, теңге.

Кесте 3 – Кәсіпорынның жылдық бірмезгілдегі шығындарын есептеу нәтижесі

Шығындар түрлері	Сумма, теңге
Оператор жалақысы	600000
Әлеуметтіксалыққа аударымдар	59400
Электрэнергияға төлем	11,48
Қорытынды:	659411,48

5.2.5 Ақпараттық өнімді енгізуден алынған үнемдеу мөлшері мен табысты есептеу

Бағдарламаны жасайтын фирма үшін табыс ретінде тапсырыс берушіге бағдарламалық өнімді сату болып есептеледі. Фирманың шығындары – ол бағдарламаны жасауға кететін шығындар және бағдарламаның көбейтілуіне кеткен шығындар. Үнемдеу мөлшерін есептейтін болсақ. Өткізу қабілетін

есептеуге кететін кәсіпорынның жалпы біржылдық шығындарын қолмен есептесек.

Қол еңбегімен мынадай шығындар есептеледі:

- жұмысшы еңбегіне төленетін ақша;
- әлеуметтік салықтар есептеледі;
- және басқа әр түрлі шығындар.
- Бір жылдық тестті қолмен құрастырушы 6 күн сағатына 1500 теңгеден жасайды.

$$6 \cdot 8 \cdot 1500 = 72000 \text{ теңге}$$

Дербес компьютер операторының жылдық жалақысы 600000 теңге

$$600000 + 72000 = 672000 \text{ теңге}$$

Әлеуметтік салыққа аударымдар (ҚР да әлеуметтік салық мөлшері 11% ҚР Салық Кодексі)

$$(672000 - 672000 \cdot 0,1) \cdot 0,11 = 66528 \text{ теңге}$$

Кесте 4 – Ақпараттық жүйені қолданбай еңбек өнімділігін талдауға кететін шығындарды есептеу

Шығындар түрлері	Сумма, теңге
Еңбекке төленетін қор	672000
Әлеуметтік салыққа аударымдар	66528
Қорытынды:	738828

Бағдарламаны қолданбай еңбек өнімділігін талдауға кететін шығындар 211008 теңге.

Бағдарламаны қолданған кездегі жылдық шығындар жоғарыда есептелді және 659411,48 теңге болды

Шығындардың шартты үнемделуі

$$\text{Эк} = 738828 - 659411,48 = 79416,5 \text{ теңге}$$

5.2.6 Пайда мен шығындарды есептеу

Пайда мен шығындарды есептеу кесте 5 те көрсетілген.

Кесте 5 - Бір жылдағы пайда мен шығындарды есептеу

Аты	Теңге
Ағымдағы шығындар, теңге	659411,48

Шартты пайда, теңге	79116,5
Пайдаға салынатын салық, теңге	9689,4
Таза пайда, теңге	69623

Кесте 6 – Ақшалай тәсілдердің қозғалысы, теңге

Аты	Жылдар		
	2014	2015	2016
Бірмезгілдік шығындар	85038,08		
Шығындарды үнемдеу		79116,5	79116,5
Ағымдағы шығындар		659411,48	659411,48
Пайдаға салынатын салық		9689,4	9689,4
Таза табыс	-85038,08	69623	69623

Кесте 6 соңы

АТЫВ	Жылдар		
	2014	2015	2016
Дисконттау коэффициенті (21% мөлшерінде)	1	0,826	0,683
Таза дисконтталған табыс (ТДТ)	-85038,08	57508,5	47552,5
ТДТөспелі нәтижесімен	-85038,08	-27529,5	20023

$$K_{д2} = 1/(1+\alpha)^1$$

$$1/(1+0,21)^1 = 0,826$$

$$K_{д3} = 1/(1+\alpha)^2$$

$$1/(1+0,21)^2 = 0,683$$

Мұндағы α – инфляция коэффициенті, $K_{д}$ - дисконттау коэффициенті
Таза дисконтталған табыс (ТДТ) = Таза табыс * Дисконттау коэффициенті
Таза дисконтталған табыс (ТДТ)

$$69623 * 0,826 = 57508,5 \text{ теңге}$$

$$69623 * 0,683 = 47552,5 \text{ теңге}$$

$$ТДТ_2 \text{ (өспелі нәтижесімен)} = ТДТ_1 + \text{Таза дисконтталған табыс (ТДТ)}_2$$

5.3 Экономикалық тиімділікті есептеу

5.3.1 Таза ағымдағы құндылықты есептеу (Net present value, NPV)

Шығындары біремізгілде жоба басында күрделі салымдар (C_0) ретінде жүзеге асатын жобалар үшін NPV төмендегі формула бойынша есептеледі:

$$NPV = \sum_{i=1}^n \frac{B_i}{(1+r)^i} - C_0.$$

мұндағы B_i – i -ші жылдағы жобадан алынатын пайда;
 r – дисконттеу мөлшері

$$NPV (21\%) = \left(\frac{69623}{(1+0,21)^1} + \frac{69623}{(1+0,21)^2} \right) - 85038,08 = 20158,2 \text{ теңге} > 0$$

5.3.2 Пайда индексін есептеу (Profitability index, PI)

PI жобаның салыстырмалы пайдасын көрсетеді. Бұл көрсеткіш келесі формуламен есептеледі

$$PI = \frac{NPV}{C_0}$$

$$PI = (105060/85038) = 1,23\%$$

5.3.3 Табыстың ішкі нормасын есептеу (Internal rate of return, IRR)

Табыстың ішкі нормасы (ТИН немесе IRR) r пайыздық мөлшерінің мәні болып есептеледі

$IRR = r$, мұндағы $NPV = 0$.

$$IRR = r_1 + \frac{f(r_1)}{f(r_1) - f(r_2)} * (r_2 - r_1)$$

$r_b = 31\%$

$PV_1 = 69623 / (1 + 0,31) = 51097,5$ теңге

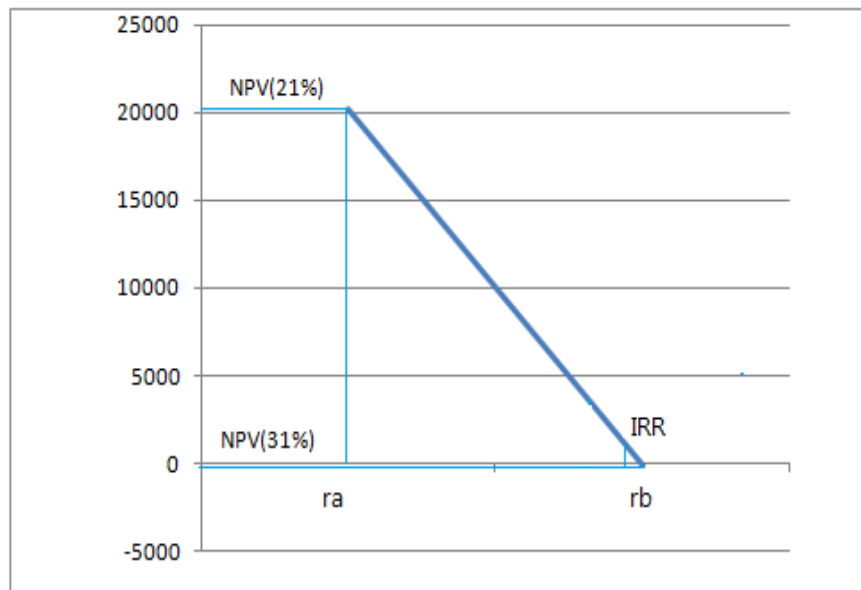
$PV_2 = 69623 / (1 + 0,31)^2 = 33901,9$ теңге

$NPV(30,0\%) = (53147 + 33901,9) - 85038,08 = -88,68$ т

$NPV(21\%) = 20158,2$ теңге

$NPV(31\%) = -88,68$ теңге

$IRR = r_a + (r_b - r_a) * NPV_a / (NPV_a - NPV_b) = 21 + (31 - 21) * 20158,2 / (20158,2 - (-88,68)) = 30,9\%$



5.3.4 Өтімділік периодын есептеу (Payback period, PBP)

Өтімділік периоды мына төмендегі формула бойынша есептеледі

$$T_{ок} = t + \frac{C_0 - \sum_{i=1}^n B_i}{B_{i+1}}$$

Мұнда:

C_0 – жоба басындағы салым;
 B_i –і-ші жылдағы жобадан алынатын табыс;
Біздің өтімділік периодымызды есептесек

$$T_{ок} = 1 + \frac{85038,08 - 57508,5}{47552,5} = 1 \text{ жыл } 7 \text{ ай}$$

Біздің жобаның өтімділік периоды 1 жыл 7 айға тең, яғни жобамыз осы уақытта өз-өзін ақтайды деген сөз.

5.4 Экономика бөлімі бойынша қорытынды

Бұл бағдарлама ағылшын тілін үйренуде таптырмас бағдарлама болып есептелді. Бұл бағдарламаны жасау үшін ақшалай қаржыландыруды керек. Бағдарламаның өзіндік құны қосымша құн салығынсыз 85038,08 теңгені қамтиды. Біздің бағдарламаны қолданған оқу орындары тест құрастыруға ақша жұмсамайды. Бағдарлама 1 жыл 7 айда өз-өзін ақтайды.

Қорытынды

Бұл дипломдық жобада барлық қойылған мақсаттар мен тапсырмалар іске асырылды. Сөздік, көп қолданылатын етістіктер, мақал-мәтелдер, тілашар үшін кестелер құрылды. Дерекқорды құру барысында барлық ерекшеліктер ескерілді.

Дерекқор MS SQL Server дерекқорларды басқару жүйесінде жасалды. Ал бағдарламаның интерфейсі Embarcadero RAD Studio Delphi 2010 және дизайны Adobe Photoshop бағдарламалық, ортасында жасалды.

Бұл бағдарламалық өнімді кез келген адам өзінің компьютеріне орнатып қолдана алады.

Тіршілік қауіпсіздігі бөлімінде бөлменің ауа алмасу (кондиционер) жүйесін есептедім. Мұнда сыртқы және ішкі микроклимат параметрлерін, температура, адам мен компьютерден бөлінетін жылу, бөлме көлемін ескере отырып плазмалық Panasonic CS/CU-E 18 NKDW сплит-жүйесі кондиционері таңдалып алынды.

Техника-экономикалық негізделуі нәтижесі бойынша бұл бағдарламаны құру ақшалай қаржыландыруды қажет етеді. Бағдарламаның өзіндік құны қосымша 85038,08 теңгені қамтиды. Бағдарлама 1 жыл 7 айда өз-өзін ақтайды.

Бұл бағдарламаның мынадай мүмкіндіктер бар:

- ағылшын тілін меңгеруді жеңілдетеді;
- ағылшын тілінен білім деңгейін тексеруге арналған тесттер;
- ағылшын тілінде ең көп қолданылатын сөздер, етістіктер;

– ағылшынша-қазақша мақал-мәтелдер, тілашар;
Бұл бағдарлама ағылшын тілін үйренушілерге көмегін тигізеді деп үміттенемін.

Пайдаланылған әдебиеттер тізімі

- 1 Никита Культин. Основы программирования в Delphi 2010. – Санкт-Петербург, 2010. – 448с.
- 2 Алексей Вишневский. SQL Server . Эффективная работа. – Санкт-Петербург, 2009. – 541с.
- 3 Никита Культин. Основы программирования в Delphi 2010. 2-е издание. – Санкт-Петербург, 2010. – 640с.
- 4 Полякова Л.Н. Основы SQL. Курс лекций: учеб. пособие. – Москва, 2004. – 368с.
- 5 Глушаков С.В., Ломотько Д.В. Базы данных: Учебный курс. – М.: ООО «Издательство АСТ», 2000. – 504с.
- 6 Ибраева Л.К., Сатимова Е.Г. Проектирование баз данных. МУ к выполнению лабораторных работ. – Алматы: АИЭС, 2004.
- 7 Дунаев В.В. Базы данных. Язык SQL. – СПб.: БХВ – Петербург, 2006.
- 8 Кандзюба С.П., Громов В.Н. Delphi 6/7. Базы данных и приложения. – СПб: ООО «ДиаСофт», 2002.
- 9 Хакимжанов Т.Е. Расчет аспирационных систем. Дипломное проектирование. Для студентов всех форм обучения всех специальностей. – Алматы: АИЭС, 2002. – 30 с.
- 10 Борисов, Е. Ф. Основы экономики: Учебное пособие / Е. Ф. Борисов. – М.: Юрайт – Издат, 2009. – 316 с.
- 11 Современная экономика: Учебное пособие /Под ред. О. Ю. Мамедова. – Ростов-на-Дону: Феникс, 2011. – 456 с.

12 Экономика: Учебное пособие /Под ред. А.С. Булатова. – М.: Юристъ, 2009. – 896 с.

13 Ерекешова З.Д., Боканова Г.Ш., 5B070400 – Есептеу техникасы және бағдарламамен қамтамасыз ету мамандығының студенттері үшін диплом жұмысының экономика бөлімін орындауға әдістемелік нұсқаулар. – Алматы: АЭЖБУ, 2014-276.

14 http://mojazarplata.kz/main/dohody-minimum/Minimalnaja_zarplata.

15 <http://www.energynews.kz/news/tarify>.

16 <http://compress.ru/article.aspx?id=11764#01>.

17 http://www.prepod2000.kulichki.net/item_286.html.

18 <http://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%80%D0%BE%D0%BA%D0%BB%D0%B8%D0%BC%D0%B0%D1%82>.

А қосымшасы

```
CREATE TABLE dictionary(  
d_Id int(4) ,  
english varchar(17) ,  
russian varchar(17),  
kazakh nchar(17),  
p_Id int ,  
v_Id int ,  
PRIMARY KEY (d_Id))  
CREATE TABLE pastsimple(  
p_Id int(4),  
baseform varchar(18),  
pastsimple varchar(18),  
pastparticiple varchar(18),  
ruskiy varchar(18),  
kazaksha nchar(18),  
PRIMARY KEY (p_Id))  
CREATE TABLE verbs(  
v_Id int(4),  
agilshin varchar(20),  
orisha varchar(20),  
kazaksha nchar(20),  
PRIMARY KEY (v_Id))  
  
CREATE TABLE makal_matelder(  

```

```
english varchar(70),
kazakh nchar(70),
m_Id int,
s_Id int,
k_Id int,
t_Id int,
su_Id int,
u_Id int,
a_Id int,
ta_Id int,
j_Id int,
ko_Id int,
ka_Id int,
sa_Id int,
sau_Id int,
me_Id int,
```

А қосымшасының жалғасы

```
ti_Id int,
PRIMARY KEY (m_id))
CREATE TABLE salemdesu(
english varchar(70),
kazakh nchar(70),
s_Id int,
PRIMARY KEY (s_id))
CREATE TABLE koshtasu(
english varchar(70),
kazakh nchar(70),
k_Id int,
PRIMARY KEY (k_id))
CREATE TABLE tanisu(
english varchar(70),
kazakh nchar(70),
t_Id int,
PRIMARY KEY (t_id))
CREATE TABLE surak(
english varchar(70),
kazakh nchar(70),
su_Id int,
PRIMARY KEY (su_id))
CREATE TABLE uakit(
english varchar(70),
kazakh nchar(70),
```



```
u_Id int,  
PRIMARY KEY (u_id))  
CREATE TABLE auaraiy(  
english varchar(70),  
kazakh nchar(70),  
a_Id int,  
PRIMARY KEY (a_id))  
CREATE TABLE tamaktanu(  
english varchar(70),  
kazakh nchar(70),  
ta_Id int,  
PRIMARY KEY (ta_id))  
CREATE TABLE jumis(  
english varchar(70),  
kazakh nchar(70),  
j_Id int,
```

А қосымшасының жалғасы

```
PRIMARY KEY (j_id))  
CREATE TABLE konak_ui(  
english varchar(70),  
kazakh nchar(70),  
ko_Id int,  
PRIMARY KEY (ko_id))  
CREATE TABLE kalada(  
english varchar(70),  
kazakh nchar(70),  
ka_Id int,  
PRIMARY KEY (ka_id))  
CREATE TABLE sayaxat(  
english varchar(70),  
kazakh nchar(70),  
sa_Id int,  
PRIMARY KEY (sa_id))  
CREATE TABLE sauda(  
english varchar(70),  
kazakh nchar(70),  
sau_Id int,  
PRIMARY KEY (sau_id))  
CREATE TABLE memleket(  
english varchar(70),  
kazakh nchar(70),  
me_Id int,  
PRIMARY KEY (me_id))
```

```
CREATE TABLE tilder(  
english varchar(70),  
kazakh nchar(70),  
ti_Id int,  
PRIMARY KEY (ti_id))
```

```
Таб толтыру  
INSERT INTO pastsimple  
VALUES('beat','beat','beaten','бит','Ұру ','7');  
INSERT INTO pastsimple  
VALUES('become','became','become','становиться','  
болу ','8');  
INSERT INTO pastsimple  
VALUES('befall','befell','befallen','случиться','болу  
,','9');
```

А қосымшасының соңы

```
INSERT INTO pastsimple  
VALUES('begird','begirt','begirt','опоясывать','буып  
алу','12');  
INSERT INTO pastsimple  
VALUES('behold','beheld','beheld','опоясывать','буып  
алу ','13');  
INSERT INTO pastsimple VALUES('bend','bent','bent;','  
зреть ','туындау ','14');  
INSERT INTO pastsimple  
VALUES('bereave','bereft;bereaved','bereft; ereaved' ,'  
наклоняться ','еңкею ','15');  
INSERT INTO pastsimple  
VALUES('beseech','besought;','besought;','лишать ','  
айрылу','16');
```

```
Сөздікте үш тілде іздеу жасауға  
SQL.Add('select * from dictionary where english like ');  
SQL.Add('select * from dictionary where russian like ');  
SQL.Add('select * from dictionary where kazakh like ');  
Сұрыптама жасауға  
'select english, russian, kazakh from dictionary'  
'select agilshin, orissha, kazaksha from verbs'  
'select english, russian, kazakh from dictionary,  
pastsimple where english=baseform'  
Және мақалдар терезесінде тілашарды шығару үшін  
төмендегідей сұраныс қолданылды 14 тақырыпқа байланысты  
14 сұраныс жазылды.  
'select english, kazakh from Salemdesu'.
```

Ә ҚОСЫМШАСЫ

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes,
  Graphics, Controls, Forms,
  Dialogs, StdCtrls, jpeg, ExtCtrls, pngimage;
type
  Form1 = class(TForm)
  Image1: TImage;
  Image2: TImage;
  Image3: TImage;
  Image4: TImage;
  Image5: TImage;
  Image6: TImage;
  Image7: TImage;
  Image8: TImage;
  Image9: TImage;
  procedure Image2Click(Sender: TObject);
  procedure Image4Click(Sender: TObject);
  procedure Image3Click(Sender: TObject);
  procedure Image5Click(Sender: TObject);
  procedure Image6Click(Sender: TObject);
  procedure Image7Click(Sender: TObject);
  procedure Image8DbClick(Sender: TObject);
  procedure Image7DbClick(Sender: TObject);
  private
    { Private declarations }
  public
```

```

    { Public declarations }
    end;
    var
    Form1: TForm1;
    implementation
    $R *.dfm}
    uses Unit2, Unit3, Unit4, Unit5, Unit6, Unit7,
unit28, unit29;
    procedure TForm1.Image2Click(Sender: TObject);
    begin
    end;
    procedure TForm1.Image3Click(Sender: TObject);
    begin
    form4.ShowModal;
    Ә қосымшасының жалғасы

    procedure TForm1.Image4Click(Sender: TObject);
    begin
    form3.ShowModal;
    end;
    procedure TForm1.Image5Click(Sender: TObject);
    begin
    form5.ShowModal;
    end;
    procedure TForm1.Image6Click(Sender: TObject);
    begin
    form6.ShowModal;
    end;
    procedure TForm1.Image7Click(Sender: TObject);
    begin
    form7.ShowModal;
    end;
    procedure TForm1.Image7DbClick(Sender: TObject);
    begin
    form29.ShowModal;
    end;
    procedure TForm1.Image8DbClick(Sender: TObject);
    begin
    form28.ShowModal;
    end;
    end.

    unit Unit3;
    interface
    uses

```

Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms,

Dialogs, StdCtrls, ExtCtrls, IniFiles, jpeg;

type

TForm3 = class(TForm)

RadioGroup1: TRadioGroup;

RadioButton1ticket: TRadioButton;

RadioButton2ticket: TRadioButton;

RadioButton3: TRadioButton;

RadioButton4: TRadioButton;

RadioButton5: TRadioButton;

RadioButton6: TRadioButton;

Ә ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

RadioButton9: TRadioButton;

RadioButton10: TRadioButton;

RadioButton11: TRadioButton;

RadioButton12: TRadioButton;

RadioButton13: TRadioButton;

RadioButton14: TRadioButton;

RadioButton15: TRadioButton;

RadioButton16: TRadioButton;

RadioButton17: TRadioButton;

RadioButton18: TRadioButton;

RadioButton19: TRadioButton;

RadioButton20: TRadioButton;

Button1starttest: TButton;

Memoquest: TMemo;

RadioButtonquest1: TRadioButton;

RadioButtonquest2: TRadioButton;

RadioButtonquest3: TRadioButton;

Button2next: TButton;

Image1: TImage;

Button1: TButton;

procedure Button1starttestClick(Sender: TObject);

procedure Button2nextClick(Sender: TObject);

procedure Button1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form3: TForm3;

Quests: TIniFile;

```

        IntnextQuest,          IntQuestTrue,          IntQuestFalse,
NumberTicket:integer;
OtvetStr, OtvetUserStr:string;
implementation
{$R *.dfm}
procedure TForm3.Button1Click(Sender: TObject);
end;
procedure          TForm3.Button1starttestClick(Sender:
TObject);
begin
IntQuestTrue:=0;

```

Ә ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```

IntQuestFalse:=0;
IntnextQuest:=1;
if RadioButton1ticket.Checked = true then
begin
// 1
NumberTicket:=1;
Memoquest.Lines.Clear;
Quests:=TIniFile.Create(ExtractFilePath(Application.ExeName)+'Quests'+IntToStr(NumberTicket)+'.ini');
Memoquest.Lines.Add(Quests.ReadString('Quest'+IntToStr(IntnextQuest), 'Quest', '.....'));

OtvetStr:=(Quests.ReadString('Quest'+IntToStr(IntnextQuest), 'Otvet', '.....'));
end
else
begin
// 2
NumberTicket:=2;
Memoquest.Lines.Clear;
Quests:=TIniFile.Create(ExtractFilePath(Application.ExeName)+'Quests'+IntToStr(NumberTicket)+'.ini');
Memoquest.Lines.Add(Quests.ReadString('Quest'+IntToStr(IntnextQuest), 'Quest', '.....'));

of
true:begin OtvetUserStr:='A'; end;
false:begin OtvetUserStr:='B'; end;
end;
if OtvetStr = OtvetUserStr then
begin
Inc(IntQuestTrue);

```

```

end
else
Inc(IntQuestFalse);
Inc(IntnextQuest);
Memoquest.Lines.Clear;
Memoquest.Lines.Clear;
Quests:=TIniFile.Create(ExtractFilePath(Application.ExeName)+'Quests'+IntToStr(NumberTicket)+'.ini');
Memoquest.Lines.Add(Quests.ReadString('Quest'+IntToStr(IntnextQuest), 'Quest', '.....'));
OtvetStr:=(Quests.ReadString('Quest'+IntToStr(IntnextQuest), 'Otvet', '.....'));
    Ә ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

    if Memoquest.Lines[0] = '.....' then
begin
    ShowMessage('Тест аяқталды!'+#13+'Дұрыс жауаптар саны:'+IntToStr(IntQuestTrue)+#13+'Дұрыс жауаптар:'+IntToStr(IntQuestFalse));
    end;
    end;
    end.

unit Unit4;
interface
uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs, jpeg, ExtCtrls, StdCtrls, Grids, DBGrids, DB, ADODB, ComObj;
type
TForm4 = class(TForm)
Image1: TImage;
Edit1: TEdit;
Edit2: TEdit;
Edit3: TEdit;
Label4: TLabel;
DBGrid1: TDBGrid;
DataSource1: TDataSource;
ADOConnection1: TADOConnection;
ADOQuery1: TADOQuery;
Button1: TButton;
Button2: TButton;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;

```

```

RadioButton3: TRadioButton;
Image2: TImage;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
procedure Edit1Change(Sender: TObject);
procedure Edit2Change(Sender: TObject);
procedure Edit3Change(Sender: TObject);
procedure Button2Click(Sender: TObject);

```

Ә ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```

procedure RadioButton1Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
end;
var
Form4: TForm4;
implementation
{$R *.dfm}
procedure TForm4.Button1Click(Sender: TObject);
var
i,j,index: Integer;
ExcelApp,sheet: Variant;
begin
ExcelApp := CreateOleObject('Excel.Application');
ExcelApp.Visible := true;
ExcelApp.WorkBooks.Add(-4167);
ExcelApp.WorkBooks[1].Worksheets[1].name := 'Otchet';
sheet:=ExcelApp.WorkBooks[1].Worksheets['Otchet'];
index:=3;
DBGrid1.DataSource.DataSet.First;
for i:=1 to DBGrid1.DataSource.DataSet.RecordCount do
begin
for j:=1 to DBGrid1.FieldCount do
sheet.cells[index,j]:=DBGrid1.fields[j-1].asString;
inc(index);
DBGrid1.DataSource.DataSet.Next;
end;
end;
procedure TForm4.Button2Click(Sender: TObject);
begin

```



```
Close;
end;
procedure TForm4.Edit1Change(Sender: TObject);
var help1 , help2: string;
begin
help1:='%'+unit4.Form4.Edit1.Text+'%';
help2:=QuotedStr(help1);
with ADOQuery1 do
begin
close;
```

Ә қосымшасының жалғасы

```
SQL.Clear;
SQL.Add('select * from dictionary where english like
'+help2);
Open;
end;
end;
procedure TForm4.Edit2Change(Sender: TObject);
var help3 , help4: string;
begin
help3:='%'+unit4.Form4.Edit2.Text+'%';
help4:=QuotedStr(help3);
with ADOQuery1 do
begin
close;
SQL.Clear;
SQL.Add('select * from dictionary where russian like
'+help4);
Open;
end;
end;
procedure TForm4.Edit3Change(Sender: TObject);
var help5 , help6: string;
begin
help5:='%'+unit4.Form4.Edit3.Text+'%';
help6:=QuotedStr(help5);
with ADOQuery1 do
begin
close;
SQL.Clear;
SQL.Add('select * from dictionary where kazakh like
'+help6);
Open;
```

```

end;
end;
procedure TForm4.RadioButton1Click(Sender: TObject);
begin
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select english, russian, kazakh
from dictionary');
ADOQuery1.Active:=True;

```

Ә қосымшасының жалғасы

```

end;
procedure TForm4.RadioButton3Click(Sender: TObject);
begin
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select english, russian, kazakh
from dictionary, pastsimple where english=baseform');
ADOQuery1.Active:=True;
end;
end.

```

```

unit Unit5;
interface
uses
Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, DB, ADODB, Grids, DBGrids, StdCtrls, comObj;
type
TForm5 = class(TForm)
DataSource1: TDataSource;
DBGrid1: TDBGrid;
ADOConnection1: TADOConnection;
ADOQuery1: TADOQuery;
Edit1: TEdit;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Button1: TButton;
Button2: TButton;
procedure Edit1Change(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);

```

```

private
{ Private declarations }
public
{ Public declarations }
end;
var
Form5: TForm5;
Ә қосымшасының жалғасы

procedure TForm5.Button1Click(Sender: TObject);
var
i,j,index: Integer;
ExcelApp,sheet: Variant;

begin
ExcelApp := CreateOleObject('Excel.Application');
ExcelApp.Visible := true;
ExcelApp.WorkBooks.Add(-4167);
ExcelApp.WorkBooks[1].Worksheets[1].name := 'Otchet';
sheet:=ExcelApp.WorkBooks[1].Worksheets['Otchet'];
index:=3;
DBGrid1.DataSource.DataSet.First;
for i:=1 to DBGrid1.DataSource.DataSet.RecordCount do
begin
for j:=1 to DBGrid1.FieldCount do
sheet.cells[index,j]:=DBGrid1.fields[j-1].asstring;
inc(index);
DBGrid1.DataSource.DataSet.Next;
end;
end;
procedure TForm5.Button2Click(Sender: TObject);
begin
Close;
end;
procedure TForm5.Edit1Change(Sender: TObject);
var help1 , help2: string;
begin
help1:='%'+unit5.Form5.Edit1.Text+'%';
help2:=QuotedStr(help1);
with ADOQuery1 do
begin
close;
SQL.Clear;
SQL.Add('select * from pastsimple where baseform like
'+help2);

```

```
Open;  
end;  
end;  
end.
```

```
unit Unit28;
```

```
Ә қосымшасының жалғасы
```

```
uses  
  Windows, Messages, SysUtils, Variants, Classes,  
  Graphics, Controls, Forms,  
  Dialogs, StdCtrls, DB, ADODB, Grids, DBGrids, jpeg,  
  ExtCtrls, comobj;  
type  
  TForm28 = class(TForm)  
    DBGrid1: TDBGrid;  
    ADOConnection1: TADOConnection;  
    ADOQuery1: TADOQuery;  
    DataSource1: TDataSource;  
    Button2: TButton;  
    Label2: TLabel;  
    Label3: TLabel;  
    Label4: TLabel;  
    Image1: TImage;  
    Button1: TButton;  
    procedure Button2Click(Sender: TObject);  
    procedure Button1Click(Sender: TObject);  
  private  
    { Private declarations }  
  public  
    { Public declarations }  
  end;  
var  
  Form28: TForm28;  
implementation  
  {$R *.dfm}  
  procedure TForm28.Button1Click(Sender: TObject);  
  var  
    i, j, index: Integer;  
  begin  
    ExcelApp := CreateOleObject('Excel.Application');  
    ExcelApp.Visible := true;  
    ExcelApp.WorkBooks.Add(-4167);  
    ExcelApp.WorkBooks[1].Worksheets[1].name := 'Otchet';  
    sheet:=ExcelApp.WorkBooks[1].Worksheets['Otchet'];
```

```

index:=3;
DBGrid1.DataSource.DataSet.First;
for i:=1 to DBGrid1.DataSource.DataSet.RecordCount do
begin
for j:=1 to DBGrid1.FieldCount do
Ә қосымшасының жалғасы

inc(index);
DBGrid1.DataSource.DataSet.Next;
end;
end;
procedure TForm28.Button2Click(Sender: TObject);
begin
Close;
end;
end.

unit Unit29;
interface
uses
Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms,
Dialogs, StdCtrls, DB, ADODB, Grids, DBGrids, ComObj,
jpeg, ExtCtrls;
type
TForm29 = class(TForm)
DBGrid1: TDBGrid;
DataSource1: TDataSource;
ADOConnection1: TADOConnection;
ADOQuery1: TADOQuery;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
RadioButton3: TRadioButton;
RadioButton4: TRadioButton;
RadioButton5: TRadioButton;
RadioButton6: TRadioButton;
Button1: TButton;
Button2: TButton;
Image1: TImage;
procedure Button2Click(Sender: TObject);
procedure RadioButton9Click(Sender: TObject);
procedure RadioButton1Click(Sender: TObject);
procedure RadioButton2Click(Sender: TObject);
procedure RadioButton12Click(Sender: TObject);
procedure RadioButton3Click(Sender: TObject);

```

```

procedure RadioButton6Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
{ Private declarations }
Ә қосымшасының соңы

end;
var
Form29: TForm29;
implementation
{$R *.dfm}
procedure TForm29.Button1Click(Sender: TObject);
var
i,j,index: Integer;
ExcelApp,sheet: Variant;
begin
ExcelApp := CreateOleObject('Excel.Application');
ExcelApp.Visible := true;
ExcelApp.WorkBooks.Add(-4167);
ExcelApp.WorkBooks[1].Worksheets[1].name := 'Otchet';
sheet:=ExcelApp.WorkBooks[1].Worksheets['Otchet'];
index:=3;
DBGrid1.DataSource.DataSet.First;
for i:=1 to DBGrid1.DataSource.DataSet.RecordCount do
begin
for j:=1 to DBGrid1.FieldCount do
sheet.cells[index,j]:=DBGrid1.fields[j-1].asString;
inc(index);
DBGrid1.DataSource.DataSet.Next;
end;
end;
procedure TForm29.Button2Click(Sender: TObject);
begin
close;
end;
procedure TForm29.RadioButton10Click(Sender:
TObject);
begin
ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select english, kazakh from
tilder');
ADOQuery1.Active:=True;
end;
end;

```

